

I/O 別冊

# FM-7/8活用研究



## FM-7/8活用研究

## CONTENTS

## システム/アプリケーション編

FM-7/8 独立モニタ	野尻 修	4
FM-7/8 音声モニタ	滝沢吉人	12
FM-7/8 SUPER MONITOR	アニー・バンネス	17
FM-8 MEMORY CHANGER	今泉裕司	22
FM-7/8 エディタ・アセンブラ Ver.2.0	馬場二郎	27
FM-7/8 FM-8 ↔ FM-7 マシン語コンバータ	コンビ	33
FM-7/8 サウンド・エディタ Ver.2.0	COMPAC YS	36
FM-7/FM-8 F-BASIC EXPANDER	笠作貴弥	57
FM-7/8 BASIC ANALYZER	平井秀明	66
FM-8 F-BASICにプロッタ・コマンドを	ROBOCON	74
FM-7/8 拡張PUT	向原 宏	81
FM-8 DOSモードDISK BASIC	樋口義人	84
FM-7/8 DISK SAVER & LOADER	中村俊之	87
FM-8 RAM FILE	渡辺満彦	93
FM-8 漢字印刷	平井正実/近藤孝吉	99
FM-7/8 濃淡スクリーン・コピー	松本忠博	107
FM-8 カラー画面コピー	JG3EC	109
FM-8 FMオシロ	滝沢吉人	114
FM-7/8 実戦 株式チャート	アイ&ケイ	119
FM-7/8 拡張Kコンパイラ	COMPAC T&S&N	126
FM-7/8 Kコンパイラ用グラフィック・サブルーチン 追加版	コンビ/茶谷公之	134
FM-7/8 スーパーパターン・エディタ	市川浩志	137
FM-7/8 日本語ワード・プロセッサ	TTC.T.S	145

## ゲーム編

FM-7/8 ALIEN FALL	正岡伸博	162
FM-7/8 キンチョマン	服部浩憲	170
FM-7 3-Dゴルフ・シミュレーション	T&Eソフト	175
FM-8 戦車最前線	高畑英樹	185
FM-7/8 3次元迷路	市川浩志	189
FM-8 FMラリー	高畑英樹	193



FM-8 日本復活	マイクロテクノロジー研究所	196
FM-8 シーソーゲーム	高畑英樹	203

## ハード編

FM-8 HERO-09の製作	笠作貴弥	210
FM-8 PSGカードの製作	COMPAC	214
FM-7 A/Dコンバータ	柏木順二	220
FM-7/8 VDPカードの製作	森田照夫	223
FM-8 リアルタイム・キースキャン	長谷 茂	236
FM-7 JOY STICKの製作	DOSのTom	238
参考資料 TLLピン接続図		240

## 資料編

FM-7サブシステム・モニタ ソース・リスト	242
FM-7 BIOSソース・リスト	276
FM-7 全回路図	289
FM-8 バブル・モードとDOSモード	308
FM-8活用研究正誤表	324

## RANDOM BOX

FM-7 ボーレート・アップ用プログラム	15
FM-8 UNLIST, Protectの解き方	
ミシン目スキップ・プログラム	21
FM-7 独立モニタ用裏RAM有効利用法	32
FM-7 BASIC ROMの有効利用法	35
FM-8 REM オプションAUTO	80
Display SubSystemの使い方	86
2 <sup>n</sup> -1計算プログラム	98
FM-7 F-BASIC Ver.1.0をFM-7で	169
コンティニューコマンドの使い方	184
拡張BASIC使用上の注意	
マシン語ダンプリストの見方	192
アセンブル・リストについて	208
サブメモリのビット・パターンを見る	222
2 dot SCROLL	235

表紙撮影協力：鈴屋ベルシャンブルサンクレスト店(03-404-5297)  
リチャード青山店(03-470-5177)



発売中！

# FM-8活用研究

## FMシリーズのルーツ

### 本書と合わせてご活用を！

IO 別冊

# FM-8活用研究

6809の特徴  
FM-8 ROMの解説

#### システム/アプリケーション編

拡張Kコンパイラ  
スクリーン・アセンブラ  
エディタ・アセンブラ  
2パス逆アセンブラ  
6809デバッグ  
6809トレーサ  
BIOS & サブシステム  
サブシステム使用法  
サブシステム転送プログラム  
Z80カードの使い方  
Z80コミュニケート  
Z80システム・モニタ  
マシン語↔配列変換  
カラーパターン・エディタ  
テレビ黒板  
グラフ作成プログラム  
Auto Plot  
フローチャート・ジェネレータ  
FM-8有象無象  
音声入出力プログラム  
騒音レベル解析  
N-BASIC→F-BASICコンバータ  
AUTO-2001 FM-8  
DISKET DOCTOR  
日本語ワード・プロセッサ

#### ゲーム編

SPACE FIGHTER  
ギャラクシアン  
ALL CAST STAR TREK  
グラフィック麻雀  
SPACE INVADERS  
パックマン  
グラフィック花札『こいこい』

#### ハード編

汎用I/Oポートの製作  
拡張インターフェイス  
PSGカード

#### 資料編

ブートストラップ・ローダと  
ディスクのフォーマット  
システム拡張モジュール  
BIOS  
ディスプレイ・サブシステム  
ソース・リスト  
全回路図集

B5判 353頁

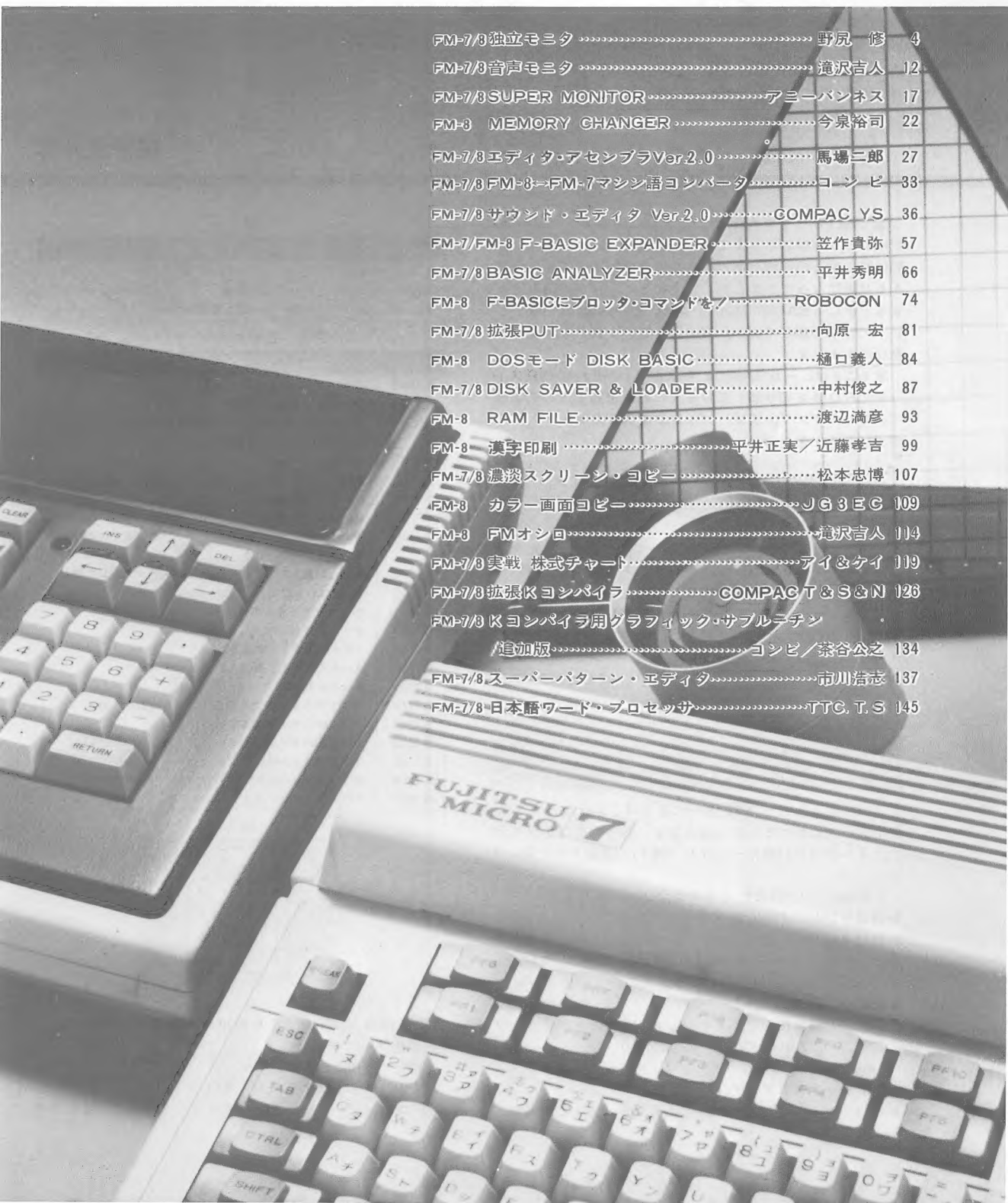
定価2500円  
(〒300)

東京・代々木

工学社

## システム/アプリケーション編

FM-7/8独立モニタ .....	野尻 修	4
FM-7/8音声モニタ .....	滝沢吉人	12
FM-7/8 SUPER MONITOR .....	アニーバンネス	17
FM-8 MEMORY CHANGER .....	今泉裕司	22
FM-7/8エディタ・アセンブラVer.2.0 .....	馬場三郎	27
FM-7/8 FM-8→FM-7マシン語コンバータ .....	コ ン ビ	33
FM-7/8サウンド・エディタ Ver.2.0 .....	COMPAC YS	36
FM-7/FM-8 F-BASIC EXPANDER .....	笠作貴弥	57
FM-7/8 BASIC ANALYZER .....	平井秀明	66
FM-8 F-BASICにプロット・コマンドを/ .....	ROBOCON	74
FM-7/8拡張PUT .....	向原 宏	81
FM-8 DOSモード DISK BASIC .....	樋口義人	84
FM-7/8 DISK SAVER & LOADER .....	中村俊之	87
FM-8 RAM FILE .....	渡辺満彦	93
FM-8 漢字印刷 .....	平井正実/近藤孝吉	99
FM-7/8濃淡スクリーン・ヨビー .....	松本忠博	107
FM-8 カラー画面ヨビー .....	J G 3 E C	109
FM-8 FMオンロ .....	滝沢吉人	114
FM-7/8実戦 株式チャート .....	アイ&ケイ	119
FM-7/8拡張Kコンパイラ .....	COMPAC T & S & N	126
FM-7/8 Kコンパイラ用グラフィック・サブルーチン 追加版 .....	ヨシビ/茶谷公之	134
FM-7/8スーパーパターン・エディタ .....	市川浩志	137
FM-7/8日本語ワード・プロセッサ .....	TTG. T. S	145

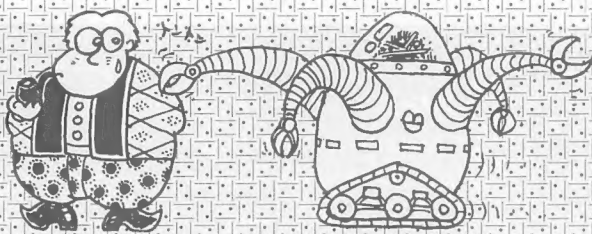






# FM-7/8 64KBオールRAMで走る

## 独立モニタ



■野尻 修

富士通FM-8を入手して半年になります。マシン語プログラム作製のネックであったBIOSについても、この間に諸先輩方の説明が進み、自在にサブシステムを利用できる状況になってきました。こうなると、マシン語を走らせるときにもBASIC ROMをつないでおかねばならないというのは場所塞ぎであり、第一眠っている裏RAMが可哀想なのであります。

本モニタは、一旦BASICからスタートさせればDIPスイッチでROMを切り離してしまうことが可能で、この状態でメモリのダンプ、変更、プリンタ制御、カセット入出力およびランチを行ないます。

### 起動方法

リストでは\$7000～\$79FFに置いてありますが、完全に位置自由です。BASICからロードし、EXEC & H7000 とするとタイトルと“PRINTER OFF”を表示してカーソルを出し、コマンド待ちになります。ROMを離すときは、本体後面のDIPスイッチ9番または10番をOFFにしてください。SPは\$7FFFにセットされますが、Mコマンドで\$7043～44を書きかえると変更できます。\$FC00～\$FC4Fがワークになっています。

### コマンド

入力とは原則として**RET**キーによる確認を行なわず、必要桁数が入力されると次へ進みます。数字入力はすべてテン・キー周辺が16進キーになり(図1)、**BS**キーが使えます。

どの時点でも**STOP**キーでコマンドに戻ります。

▶Wコマンド：画面桁数を40↔80に変更。

▶Dコマンド：メモリ・ダンプ。開始番地と終了番地を入力するとリストを出し始め、**STOP**キーで終了、**ESC**キーで一時的停止(ただし**ESC**がかかっている間は**STOP**が働かない)。

DD——16×16バイト・チェック・サム付き16進ダンプとASCIIダンプを並べて出力。

DS——8×8バイト・チェック・サム付き16進ダンプ。

DA——ASCIIダンプ

▶Mコマンド：メモリ変更。開始番地を入力すると番地、

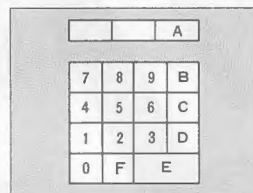
表1

アドレス	内 容
\$7020, \$7046	FC ワーク・ページ
\$7043～44	7FFF SPの値
\$70ED～EF	7FFF BASIC復帰時のSP

表2 主なサブルーチン

アドレス	内 容
\$71DB	CRT 1字出力
\$71EE	プリンタ 1字出力 以下は\$FC0Eのビット7が1のときCRTとプリンタ、0のときCRTのみに出力
\$71CF	1字出力 (AccA)
\$7176	改行
\$7249	AccAを16進表示
\$7259	X regを16進表示
\$7321	X regの番地からAccB個をキャラクタとして表示(制御コードはピリオド)
\$733B	X regの番地から0があるまでキャラクタとして表示(制御コード有効)
\$71AC	キーボード 1字入力 (AccA)
\$71A2	キー入力を持ってAccAに取る
\$718A	1字キー入力+1字出力
\$7229	AccAの1バイト数→ASCIIコード(D)
\$7265	AccAのASCIIコード→数値(B)
\$7273	16進1桁のキー入力を数値にしてAccBに入れる
\$72C5	16進2桁のキー入力をAccDに入れる
\$72C9	16進4桁のキー入力をAccDに入れる
\$7205	サブシステムHALT, FIRQマスク
\$721E	サブシステムRESTART, FIRQマスクを元に戻す
\$79E0	1行入力。Kコンパイル用におまけ

図1



メモリ内容、キャラクタを表示し、変更を可能にする。

**STOP**キーで終了。

MN——16進数入力。スペース・バーでスキップ。

MC——キャラクタ入力。**RET**キーでスキップ。

▶Fコマンド：メモリを同一データでクリア。開始番地、

最終番地、16進データを入力。

コマンド F, V, G, C, Bはデータ入力後“ok? (Y)”と確認してくるので、[Y]を入力。他のキーではコマンドに戻る。

▶Vコマンド：メモリ・ブロック移動。先頭番地、最終番地、目的地の先頭番地の順に入力。元の位置と重なり合う領域への移動もOK。

▶Gコマンド：ブランチ。実行開始番地を入力するとFIRQを許可して分岐する。実行中[STOP]キーでコマンドに戻る。プログラムの終わりはRTS(\$39)、SWI(\$3F)、SWI3(\$113F)のいずれでもコマンドに戻る。SWIまたは[ST

OP]キーで戻った場合、レジスタがCC~PCの順でワーク・ページ先頭(\$FC00~\$FC0B)に格納される。

▶Cコマンド：ワーク・ページ先頭の内容をレジスタに入れて、PCの値(\$FC0A~0B)へ分岐。他はGコマンドと同じ。

▶Rコマンド：レジスタを格納した\$FC00~0Bの値を変更。スペース・バーでスキップ。

▶Pコマンド：プリンタのレディ・チェックをして、プリンタ・フラグをON/OFF。

▶Lコマンド

LD-カセットテープのマシン語データをロード。ファイル名とロード開始番地を入力してください。8文字または[RET]が入力されるまでの文字列がファイル名になる。

L? — テープ上のサムをチェック。ファイル名のみ入力。

▶Sコマンド：カセットテープへのセーブ。ファイル名、先頭番地、最終番地、実行開始番地を入力すると、F-BASICと共通に使えるテープを作る。

▶Hコマンド：コマンドのリストを表示。

▶Bコマンド：システム・ベクトルを直し、SPを\$7FFFにセットしてBASICのモニタへジャンプ。当然ながらROMが離れているときに実行すると制御不能になる。

## 停止方法

①モニタとSPを\$7000以下に置いて、DIPスイッチ9番、10番をONにしてBコマンドでBASICのモニタを出す。

②DIPスイッチ切り換えを強行し、RESET。

## プログラムの説明

### 独立モニタ アセンブル・リスト

```

0001 0000 ; FMB INDEPENDENT MONITOR
0002 0000 ; Version 1.0
0003 0000 ; Coded by O. NOJIRI
0004 0000 ;
0005 0000 PAGE: EQU $FC
0006 0000 MN: EQU $00
0007 0000 SKF: EQU $00
0008 0000 PRF: EQU $0E
0009 0000 WID: EQU $0F
0010 0000 AST: EQU $0D
0011 0000 CST: EQU $0C
0012 7000 ORG $7000
0013 7000 2009 BRA RESET
0014 7002 1600EF LBRA FIRQ
0015 7005 160113 LBRA REGST
0016 7008 160036 LBRA RESET0
0017 7008 1F50 RESET: TFR PC,0
0018 700D 830000 SUBD #11
0019 7010 F0FFF6 STD $FFF6
0020 7013 C30003 ADDD #3
0021 7016 F0FFFA STD $FFFA
0022 7019 C30003 ADDD #3
0023 701C F0FFF2 STD $FFF2
0024 701F 86FC LDA #PAGE

```

ROMを離すとRESET, STOP+RESETが使えないので、ストップ・キーを押すといつでもコマンドに戻るようになっています。各コマンドは\$704F~70AFのループから単純にジャンプし、SWI3で終了しています。SWI2は未使用で、SWIおよびストップ・キーのFIRQではレジスタを格納してコマンドに戻ります。その他のFIRQは\$70FEから\$79FDに飛びます。RTI(\$3B)になっているので変更して活用してください。ユーザープログラムでサブシステムがHALTされているときにFIRQがかかると制御不能になるので、HALT, RESTARTはモニタ内ルーチンを使う方が良いでしょう。

主な定数とサブルーチンを表1, 2に示します。

## 裏RAMの異常な利用法

何かプログラムを\$8000以降に転送しておき、ROMをつないでRESET。モニタを入れ直して裏RAMを見ると、プログラムはちゃんと残っているのです。

FM-7でもROM/RAMの切り換えはディップ・スイッチでよいわけですが、ソフトでも切り換えができます。この切り換えのルーチンを作り、\$70A8以降の余白から飛ばすと便利です。なお、この最後はSWI3(\$11, \$3F)としてください。

カセット入出力については、高速モードで使う場合、ボーレートも高くなってしまいます(個人で使っている場合はこの方がいいかも?)。そこで、TIMERルーチンと呼ぶときの値(\$7755, \$77A6, \$77B1, \$77D3, \$7803, \$780A, \$7811, \$7818, \$7826, \$782D, \$7834, \$783B, \$7850, \$7857, \$7864の各2バイト)を変えてください。

また、94行めのLDD #CC57をLDD #C953に、108行めのJMP \$AFA2をJMP \$ABF4に変更するとBコマンドが動作します。

DISKで使う場合、システム・スタックの設定値(\$7043, 44と\$70EE, EF)をたとえば\$6FFFに変更すればOKです。

このプログラムと本誌の『エディタ・アセンブラ』をリンクして使うと便利です。ぜひ一度試してみてください。

### 参考文献

- 1) FM-8活用研究の各記事
- 2) F-BASIC文法書
- 6) 6809ハンドブック

```

0025 7021 1F88 TFR A,DP
0026 7023 0F0E CLR <PRF
0027 7025 8628 LDA #40
0028 7027 970F STA <WID
0029 7029 170100 LBSR CONS
0030 702C C600 LDB #0 ; Title
0031 702E 1705A8 LBSR DPCX
0032 7031 C625 RESET1:LDB #25 ; LF+PRINTER
0033 7033 1705A6 LBSR DPCX
0034 7036 C633 LDB #33 ; 'OFF'
0035 7038 960E LDA <PRF
0036 703A 2702 BEQ RST1
0037 703C C630 LDB #30 ; 'ON'
0038 703E 17059B RST1: LBSR DPCX
0039 7041 10CE7FFF RESET0:LDB $7FFF
0040 7045 86FC LDA #PAGE
0041 7047 1F8B TFR A,DP
0042 7049 960E LDA <PRF
0043 704B 847F ANDA #7F ; Printer STOP
0044 704D 970E STA <PRF
0045 704F 170124 CHD0: LBSR LIN
0046 7052 863E LDA #3E
0047 7054 170178 LBSR PRINT
0048 7057 170148 LBSR INKEY

```

## 独立モニタ アセンブル・リスト

0049 705A 1CBF	ANDC #5BF	; FIRO unmask	0153 7150 ED81	STD ,X++
0050 705C 8142	CMPA #542		0154 7152 C600	LDB #0
0051 705E 275F	BEQ EXIT		0155 7154 E784	STB ,X
0052 7060 8144	CMPA #544		0156 7156 1700C5	LBSR RSTR
0053 7062 102702E2	LBEQ DAMP		0157 7159 1700A9	LBSR HLT
0054 7066 8157	CMPA #557		0158 715C CC0C13	LDD #0C13
0055 7068 102700FA	LBEQ WIDTH		0159 715F F0FC82	STD #FC82
0056 706C 8140	CMPA #540		0160 7162 1700B9	LBSR RSTR
0057 706E 10270431	LBEQ MEM		0161 7165 39	RTS
0058 7072 8146	CMPA #546		0162 7166	; WIDTH CHANGE
0059 7074 102704D2	LBEQ FILL		0163 7166 960F	WIDTH: LDA <WID
0060 7078 8147	CMPA #547		0164 7168 C650	LDB #00
0061 707A 102704A0	LBEQ JUMP		0165 716A 8128	CMPA #40
0062 707E 8156	CMPA #556		0166 716C 2702	BEQ WID0
0063 7080 102704EC	LBEQ MOVE		0167 716E C620	LDB #40
0064 7084 8150	CMPA #550		0168 7170 070F	WID0: STB <WID
0065 7086 10270528	LBEQ LPT		0169 7172 80C3	BSR CONS
0066 708A 814C	CMPA #54C		0170 7174 113F	FDB #113F ; SW13
0067 708C 1027086F	LBEQ LOAD		0171 7176	; LINE FEED
0068 7090 8153	CMPA #553		0172 7176 3403	LIN: PSHS A,CC
0069 7092 102705FC	LBEQ SAVE		0173 7178 860A	LDA #4A
0070 7096 8140	CMPA #548		0174 717A 8053	BSR PRINT
0071 7098 10270539	LBEQ HELP		0175 717C 8600	LDA #50
0072 709C 8143	CMPA #543		0176 717E 804F	BSR PRINT
0073 709E 102704B8	LBEQ CALL		0177 7180 3583	PULS A,CC,PC
0074 70A2 8152	CMPA #552		0178 7182	; PRINT A SPACE
0075 70A4 10270461	LBEQ REG		0179 7182 3403	SPC: PSHS A,CC
0076 70A8 12	NOP		0180 7184 8620	LDA #620
0077 70AA 12	NOP		0181 7186 8047	BSR PRINT
0078 70AC 12	NOP		0182 7188 3583	PULS A,CC,PC
0079 70AD 12	NOP		0183 718A	; TAKE KEY IN A REG AND DISPLAY
0080 70AE 12	NOP		0184 718A 3401	INPUT: PSHS CC
0081 70B0 12	NOP		0185 718C 0F00	CLR <SKF
0082 70BA 209F	BRA CHD0		0186 718E 8012	INPUT0:BSR INKEY
0083 70BC C637	OKCHCK: LDB #537		0187 7190 8100	CMPA #50
0084 70BE 170527	LBSR DPCX		0188 7192 2700	BEQ INPUT1
0085 70C0 170002	LBSR INPUT		0189 7194 8120	CMPA #520
0086 70C2 8159	CMPA #559		0190 7196 2502	BOS INPUT2
0087 70C4 2702	BEQ OKC1		0191 7198 8035	BSR PRINT
0088 70C6 113F	FDB #113F		0192 719A 3581	INPUT2:PULS CC,PC
0089 70C8 39	RTS		0193 719C 8AFF	INPUT1:LDB #5FF
0090 70CA 170100	EXIT: LBSR PRINT		0194 719E 0700	STB <SKF
0091 70CC D0EC	BSR OKCHCK		0195 71A0 28F8	BRA INPUT2
0092 70CE 867E	LDA #57E		0196 71A2	; WAIT KEY & TAKE IN A REG
0093 70D0 8701E0	STA #01E0		0197 71A2 3401	INKEY: PSHS CC
0094 70D2 CCCC57	LDD #CCCC57		0198 71A4 8006	INKEY0:BSR KEY
0095 70D4 F001E1	STD #01E1		0199 71A6 8100	CMPA #0
0096 70D6 8638	LDA #838		0200 71A8 27FA	BEQ INKEY0
0097 70D8 8701D1	STA #01D1		0201 71AA 3581	PULS CC,PC
0098 70DA 8701D7	STA #01D7		0202 71AC	; KEY CODE INTO A REG
0099 70DC CC01E0	LDD #001E0		0203 71AC 3405	KEY: PSHS B,CC
0100 70DE F0FFFA	STD #FFFA		0204 71AE 8055	LDB HLT
0101 70E0 CC01D7	LDD #001D7		0205 71B0 CC2900	LDD #2900
0102 70E2 F0FFFA	STD #FFFA		0206 71B3 F0FC82	STD #FC82
0103 70E4 CC01D1	LDD #001D1		0207 71B6 8046	BSR RSTR
0104 70E6 F0FFFA	STD #FFFA		0208 71B8 8048	BSR HLT
0105 70E8 4F	CLRA		0209 71BA C608	LDB #508
0106 70EA 1F88	TFR A,D,P		0210 71BC 86FC83	LDA #FC83
0107 70EC 10CE7FFF	LDS #47FFF		0211 71BF 86FC80	ORD #FC80
0108 70FE 7EAF82	JMP #AFA2		0212 71C2 F7FC80	STB #FC80
0109 70F3 12	NOP		0213 71C5 8057	BSR RSTR
0110 70F4 9700	FIRO: STA <AST		0214 71C7 3585	PULS B,CC,PC
0111 70F6 8602	LDA #2		0215 71C9	; DISP B CHRS IN A
0112 70F8 85F084	BITA #F084		0216 71C9 8004	STRING:BSR PRINT
0113 70FA 102608FC	LBNE TIMIR		0217 71CB 5A	DECB
0114 70FF 8602	LDA #2		0218 71CC 26FB	BNE STRING
0115 7101 8AF084	ORA #F084		0219 71CE 39	RTS
0116 7104 87F084	STA #F084		0220 71CF	; DISP ASCII IN A REG
0117 7107 86C0	LDA #8C0	; Motor off	0221 71CF 800A	PRINT: BSR PRINT
0118 7109 87F080	STA #F080		0222 71D1 3405	PSHS B,CC
0119 710C 3502	PULS A	; CC in A	0223 71D3 060E	LDB <PRF
0120 710E 970C	STA <CST	; CC in memory	0224 71D5 2A02	BPL PRINT0
0121 7110 9600	LDA <AST	; A back	0225 71D7 8015	BSR LPRINT
0122 7112 347E	PSHS U,Y,X,DP,B,A		0226 71D9 3585	PRINT0:PULS B,CC,PC
0123 7114 960C	LDA <CST	; CC in A	0227 71DB 3431	TPRINT:PSHS X,Y,CC
0124 7116 3402	PSHS A		0228 71DD 8026	BSR HLT
0125 7118 170091	LBSR KEY		0229 71DF 108EFC82	LDY #5FC82
0126 711B 1F88	REGST: TFR DP,A		0230 71E3 8E0301	LDX #0301
0127 711D C600	LDB #MM		0231 71E6 AFA1	STX ,Y++
0128 711F 1F81	TFR D,X		0232 71E8 A7A4	STA ,Y
0129 7121 C60C	LDB #12		0233 71EA 8032	BSR RSTR
0130 7123 3502	FIRO2: PULS A		0234 71EC 3581	PULS Y,X,CC,PC
0131 7125 A700	STA ,X+		0235 71EE 3405	LPRINT:PSHS B,CC
0132 7127 5A	DECB		0236 71F0 F6FD02	LPI: LDB #FD02
0133 7128 26F9	BNE FIRO2		0237 71F3 54	LSRB
0134 712A 16FF14	LBRA RESET0		0238 71F4 25FA	BOS LPI
0135 712D	; TEMPORARY LPT FLAG		0239 71F6 87FD01	STA #FD01
0136 712D 3403	LPTON: PSHS A,CC		0240 71F9 C600	LDB #0
0137 712F 960E	LDA <PRF		0241 71FB F7FD00	STB #FD00
0138 7131 48	ASLA		0242 71FE CA40	ORD #540
0139 7132 47	ASRA		0243 7200 F7FD00	STB #FD00
0140 7133 970E	STA <PRF		0244 7203 3585	PULS B,CC,PC
0141 7135 3583	PULS A,CC,PC		0245 7205	; SUB SYSTEM HALT
0142 7137	; CONSOLE & WIDTH		0246 7205 3402	HLT: PSHS A
0143 7137 1700CB	CONS: LBSR HLT		0247 7207 1FA8	TFR CC,A
0144 713A 8EFC82	LDX #8EFC82		0248 7209 970C	STA <CST
0145 713D CC0100	LDD #00100		0249 720B 86FD05	HLT0: LDA #FD05
0146 7140 ED81	STD ,X++		0250 720E 28FB	BMI HLT0
0147 7142 960F	LDA <WID		0251 7210 1A40	ORCC #540
0148 7144 C619	LDB #519		0252 7212 8600	LDA #500
0149 7146 ED81	STD ,X++		0253 7214 87FD05	STA #FD05
0150 7148 CC0019	LDD #00019		0254 7217 86FD05	HLTI: LDA #FD05
0151 714B ED81	STD ,X++		0255 721A 2AF8	BPL HLT1
0152 714D CC0001	LDD #00001		0256 721C 3582	PULS A,PC

; FIRO MASK



```

0257 721E      ; SUB SYSTEM RESTART
0258 721E 3402  RESTRT:PSHS A
0259 7220 7FFD05 CLR $FDD5
0260 7223 9608 LDA KCST
0261 7225 1F8A TFR A,CC
0262 7227 3582 PULS A,PC
0263 7229      ; BYTE IN A TO ASCII IN A&B
0264 7229 3401 ENCODE:PSHS CC
0265 722B 44 LSR A
0266 722C 56 RORB
0267 722D 44 LSR A
0268 722E 56 RORB
0269 722F 44 LSR A
0270 7230 56 RORB
0271 7231 44 LSR A
0272 7232 56 RORB
0273 7233 54 LSR B
0274 7234 54 LSR B
0275 7235 54 LSR B
0276 7236 54 LSR B
0277 7237 8B30 ADDA #530
0278 7239 813A CHPA #53A
0279 723B 2582 BCS ENC1
0280 723D 8B07 ADDA #7
0281 723F C830 ENC1: ADDB #530
0282 7241 C13A CHPB #53A
0283 7243 2582 BCS ENC2
0284 7245 C807 ADDB #7
0285 7247 35B1 ENC2: PULS CC,PC
0286 7249      ; DISP A REG DATUM
0287 7249 3407 BYTEDP:PSHS A,B,CC
0288 724B 80DC BSR ENCODE
0289 724D 17FF7F LBSR PRINT
0290 7250 1E89 EIG A,B
0291 7252 17FF7A LBSR PRINT
0292 7255 1E89 EIG A,B
0293 7257 35B7 PULS A,B,CC,PC
0294 7259      ; DISP X REG
0295 7259 3407 DPRX: PSHS A,B,CC
0296 725B 1F10 TFR X,D
0297 725D 8DEA BSR BYTEDP
0298 725F 1F9B TFR B,A
0299 7261 8DE6 BSR BYTEDP
0300 7263 35B7 PULS A,B,CC,PC
0301 7265      ; ASCII IN A TO NUMBER IN B
0302 7265 3401 DECODE:PSHS CC
0303 7267 1F89 TFR A,B
0304 7269 C830 SUBB #530
0305 726B C10A CHPB #5A
0306 726D 2582 BCS DEC0
0307 726F C807 SUBB #7
0308 7271 35B1 DEC0: PULS CC,PC
0309 7273      ; TAKE 8 BIT No KEY TO B
0310 7273 3403 GETNB0:PSHS A,CC
0311 7275 17FF2A G8B0: LBSR INKEY
0312 727B 0F0D CLR <SKF
0313 727A B108 CHPA #8
0314 727C 273D BEQ GK1
0315 727E B120 CHPA #520
0316 7280 273D BEQ GK3
0317 7282 B11C CHPA #51C
0318 7284 2718 BEQ GKCA
0319 7286 B12C CHPA #52C
0320 7288 2718 BEQ GKCB
0321 728A B12B CHPA #52B
0322 728C 2718 BEQ GKCC
0323 728E B12D CHPA #52D
0324 7290 2718 BEQ GKCD
0325 7292 B10D CHPA #5D
0326 7294 2718 BEQ GKCE
0327 7296 B12E CHPA #52E
0328 7298 2616 BNE GKC
0329 729A B646 LDA #546
0330 729C 2012 BRA GKC
0331 729E B641 GKCA: LDA #541
0332 72A0 200E BRA GKC
0333 72A2 B642 GKCB: LDA #542
0334 72A4 200A BRA GKC
0335 72A6 B643 GKCC: LDA #543
0336 72A8 2006 BRA GKC
0337 72AA B644 GKCD: LDA #544
0338 72AC 2002 BRA GKC
0339 72AE B645 GKCE: LDA #545
0340 72B0 80B3 GKC: BSR DECODE
0341 72B2 C10F CHPB #5F
0342 72B4 228F BHI G8B0
0343 72B6 17FF16 LBSR PRINT
0344 72B9 35B3 GKC2: PULS A,CC,PC
0345 72BB C6F0 GK1: LDB #5F0
0346 72BD 20FA BRA GK2
0347 72BF C6FF GK3: LDB #5FF
0348 72C1 D70D STB <SKF
0349 72C3 20FA BRA GKC2
0350 72C5      ; TAKE 1 BYTE No KEY TO B
0351 72C5 C602 GETN1: LDB #2
0352 72C7 2002 BRA GETN
0353 72C9      ; TAKE 2 BYTE No KEY TO B
0354 72C9 C604 GETN2: LDB #4
0355 72CB 3431 GETN: PSHS X,Y,CC
0356 72CD 327E LEAS -2,S
0357 72CF 17FE80 LBSR SPC
0358 72D2 6FE4 CLR ,S
0359 72D4 1F9B TFR B,A
0360 72D6 1F01 GNB: TFR D,X

```

FM-7の場合次のところを変更

```

72B7 2C=> 2D
72BB 2B=> 3D
72BF 2D=> 2C

```

; when key = BS

; when key = SPACE

```

0361 72DB 8D99 BSR GETNB8
0362 72DA 1F02 TFR D,Y
0363 72DC C1F0 CHPB #5F0
0364 72DE 2723 BEQ GN1
0365 72E0 C1FF CHPB #5FF
0366 72E2 271B BEQ GN3
0367 72E4 ECE4 LDD ,S
0368 72E6 58 LSLB
0369 72E7 49 ROLA
0370 72E8 58 LSLB
0371 72E9 49 ROLA
0372 72EA 58 LSLB
0373 72EB 49 ROLA
0374 72EC 58 LSLB
0375 72ED 49 ROLA
0376 72EE EDE4 STD ,S
0377 72F0 1F20 TFR Y,D
0378 72F2 8600 LDA #0
0379 72F4 E3E4 ADDD ,S
0380 72F6 EDE4 STD ,S
0381 72F8 1F10 TFR X,D
0382 72FA 5A DECB
0383 72FB 26D9 BNE GN0
0384 72FD ECE4 LDD ,S
0385 72FF 3262 GN3: LEAS 2,S
0386 7301 35B1 PULS X,Y,CC,PC
0387 7303 1F10 GN1: TFR X,D
0388 7305 A77F STA -1,S
0389 7307 E17F CHPB -1,S
0390 7309 27CB BEQ GN0
0391 730B C66E LDB #BS
0392 730D 1702CC LBSR DPCX
0393 7310 ECE4 LDD ,S
0394 7312 44 LSR A
0395 7313 56 RORB
0396 7314 44 LSR A
0397 7315 56 RORB
0398 7316 44 LSR A
0399 7317 56 RORB
0400 7318 44 LSR A
0401 7319 56 RORB
0402 731A EDE4 STD ,S
0403 731C 1F10 TFR X,D
0404 731E 5C INCB
0405 731F 20B5 BRA GN0
0406 7321      ; DISP B BYTE CHR FROM X
0407 7321 3403 DPC: PSHS A,CC
0408 7323 A6B0 DPC0: LDA ,X+
0409 7325 81FF CHPA #5FF
0410 7327 270B BEQ DC1
0411 7329 817F CHPA #57F
0412 732B 2704 BEQ DC1
0413 732D 811F CHPA #51F
0414 732F 2202 BHI DC0
0415 7331 862E DC1: LDA #2E
0416 7333 17FE99 DC0: LBSR PRINT
0417 7336 5A DECB
0418 7337 26EA BNE DPC0
0419 7339 35B3 PULS A,CC,PC
0420 733B      ; DISP CHR FROM X TILL 0
0421 733B 3403 DPCX0: PSHS A,CC
0422 733D A6B0 DPCX0: LDA ,X+
0423 733F 2705 BEQ DPX1
0424 7341 17FE8B LBSR PRINT
0425 7344 20F7 BRA DPX0
0426 7346 35B3 DPX1: PULS A,CC,PC
0427 7348      ;
0428 7348      ; DAMP
0429 7348      ;
0430 7348 17FE84 DAMP: LBSR PRINT
0431 734B 32E8EA LEAS -22,S
0432 734E 17FE51 DMP0: LBSR INKEY
0433 7351 8141 CHPA #541
0434 7353 270D BEQ ADAMP
0435 7355 8144 CHPA #544
0436 7357 270E BEQ DDAMP
0437 7359 8153 CHPA #553
0438 735B 26F1 BNE DMP0
0439 735D 8E000B LDX #0
0440 7360 200B BRA DMP1
0441 7362 8E0220 ADAMP: LDX #50220
0442 7365 2003 BRA DMP1
0443 7367 8E0110 DDAMP: LDX #50110
0444 736A AFE90011 DMP1: STX #11,S
0445 736E 17FE5E LBSR PRINT
0446 7371 17FF55 LBSR GETNB2
0447 7374 A7E90013 STA #13,S
0448 7378 A6E90011 LDB #11,S
0449 737C 54 LSRB
0450 737D 54 LSRB
0451 737E 54 LSRB
0452 737F 8100 CHPA #0
0453 7381 270B BEQ DMP3
0454 7383 54 LSRB
0455 7384 8101 CHPA #1
0456 7386 2702 BEQ DMP2
0457 7388 54 LSRB
0458 7389 58 ASLB
0459 738A 58 ASLB
0460 738B 58 ASLB
0461 738C 58 ASLB
0462 738D 58 ASLB
0463 738E E7E90014 STB #14,S
0464 7392 17FF34 LBSR GETNB2

```

; LF+ 'Sum:

: SW13

## 独立モニタ アセンブル・リスト

0673	756A ACE4	CMPX ,S				0769	7655 0A00	FDB \$0A00	
0674	756C 25FA	BCS FILL1				0770	7657 5368697020	FCC /Skip	
0675	756E 113F	FDB \$113F			; SW13				
0676	7570					0771	765D 00	FCB 0	
0677	7570	: MOVE				0772	765E 0020	FDB \$0020	
0678	7570					0773	7660 0000	FDB \$0000	
0679	7570 17FC5C	MOVE: LBSR PRINT				0774	7662 0A00	FDB \$0A00	
0680	7573 327A	LEAS -6,S				0775	7664 5B57204444	FCC /LW D0 DS DA MN MC F /	
0681	7575 17FD51	LBSR GETN2B							
0682	7578 EDE4	STD ,S							
0683	757A 17FD4C	LBSR GETN2B							
0684	757D E062	STD ,S				0776	7678 5220472050	FCC /R G P LD L? S B3/	
0685	757F 17FD47	LBSR GETN2B							
0686	7582 ED64	STD ,S							
0687	7584 17FB29	LBSR DKCHCK							
0688	7587 EC64	MOVE1: LDD ,S				0777	7680 00	FCB 0	
0689	7589 10A3E4	CMPD ,S				0778	7689	RMB 9	
0690	758C 220F	BHI MOVE2				0779	7692		
0691	7590 AEE4	LDX ,S				0780	7692		; CASSETTE MT INTERFACE
0692	7590 10AE64	LDY ,S				0781	7692		
0693	7593 A600	MOVE3: LDA ,X+				0782	7692		
0694	7597 A7A0	STA ,Y+				0783	7692		SH: EDU \$65
0695	7597 AC62	CMPX ,S				0784	7692		LD: EDU \$5C
0696	7599 23F0	BLS MOVE3				0785	7692		BS: EDU \$6E
0697	759B 113F	FDB \$113F			; SW13	0786	7692		ES: EDU \$40
0698	759D AE62	MOVE2: LDX ,S				0787	7692		TYPE: EDU \$10
0699	759F EC62	LDD ,S				0788	7692		SUM: EDU \$11
0700	75A1 A3E4	SUBD ,S				0789	7692		LEN: EDU \$12
0701	75A3 E364	ADD0 ,S				0790	7692		ADDRESS: EDU \$13
0702	75A5 1F02	TFR D,Y				0791	7692		ENDAD: EDU \$15
0703	75A7 3001	LEAX 1,X				0792	7692		CPF: EDU \$17
0704	75A9 3121	LEAY 1,Y				0793	7692		FOUND: EDU \$18
0705	75AB A602	MOVE4: LDA ,X				0794	7692		VF: EDU \$19
0706	75AD A7A2	STA ,Y				0795	7692		NAME: EDU \$1A
0707	75AF ACE4	CMPX ,S				0796	7692		DATA: EDU \$22
0708	75B1 22F0	BHI MOVE4				0797	7692		DATEND: EDU \$27
0709	75B3 113F	FDB \$113F			; SW13	0798	7692		HBAD: EDU \$2C
0710	75B5					0799	7692		DATHS: EDU \$2C
0711	75B5					0800	7692		DATENA: EDU \$2E
0712	75B5					0801	7692		HB: EDU \$30
0713	75B5 17FC17	LPT: LBSR PRINT				0802	7692		
0714	75B8 0600	LDA 00				0803	7692		; SAVE
0715	75BA D60E	LOB <PRF				0804	7692		
0716	75BC 2612	BNE LPT0				0805	7692 17FB3A		SAVE: LBSR PRINT
0717	75BE 8AFD02	LDA \$FD02				0806	7695 170303		LBSR NAMEIN
0718	75C1 040A	ANDA \$0A				0807	7698 1F00		TFR DP,A
0719	75C3 0102	CMPA #2				0808	769A C627		LDL <DATH+5
0720	75C5 2707	BEQ LPT1				0809	769C D02C		STD <DATHS
0721	75C7 C640	LOB #ER				0810	769E C627		LDL <DATEND
0722	75C9 170010	LBSR DPCX				0811	76A0 D02E		STD <DATENA
0723	75CC 113F	FDB \$113F			; SW13	0812	76A2 C61A		LDL <NAME
0724	75CE 067F	LPT1: LDA #67F				0813	76A4 1F01		TFR D,X
0725	75D0 970E	LPT0: STA <PRF				0814	76A6 C630		LDL #HB
0726	75D2 16FA5C	LBRA RESET1				0815	76A8 1F02		TFR D,Y
0727	75D5					0816	76AA 17FC1C		LBSR GETN2B
0728	75D5 C672	HELP: LDB #472			; Menu	0817	76AD D013		STD <ADDRESS
0729	75D7 170002	LBSR DPCX				0818	76AF D025		STD <DATH+3
0730	75DA 113F	FDB \$113F			; SW13	0819	76B1 17FC15		LBSR GETN2B
0731	75DC					0820	76B4 C30001		ADD0 #1
0732	75DC 3413	DPCX: PSHS X,A,CC				0821	76B7 D015		STD <ENDAD
0733	75DE 0600	LDA 00				0822	76B9 9313		SUBD <ADDRESS
0734	75E0 ED7E	STD -2,S				0823	76BB D023		STD <DATH+1
0735	75E2 1F50	TFR PC,D				0824	76BD 17FC09		LBSR GETN2B
0736	75E4 E37E	ADD0 -2,S				0825	76C0 D02A		STD <DATEND+3
0737	75E6 C3000C	ADD0 #3C				0826	76C2 C600		LDL #0
0738	75E9 1F01	TFR D,X				0827	76C4 A600		SAVE0: LDA ,X+
0739	75EB 17FD4D	LBSR DPCX0				0828	76C6 A7A0		STA ,Y+
0740	75EE 3593	PULS X,A,CC,PC				0829	76C8 5A		DECB
0741	75F0 464D2D3020	FCC /FM-B /				0830	76C9 26F9		BNE SAVE0
0742	75F5 494E444550	FCC /INDEPENDENT /				0831	76CB 0602		LDA #2
	454E44454E					0832	76CD 9730		STA <HB+8
	5420					0833	76CF 06FF		LDA #5FF
0743	7601 4D4F4E4954	FCC /MONITOR /				0834	76D1 9727		STA <DATEND
	4F5220					0835	76D3 1F00		TFR DP,A
0744	7609 5665727369	FCC /Version 1.0/				0836	76D5 C639		LDL #HB+9
	6F6E20312E					0837	76D7 1F01		TFR D,X
	30					0838	76D9 0600		LDA #0
0745	7614 00	FCB 0				0839	76DB 9722		STA <DATH
0746	7615 0A00	FDB \$0A00				0840	76DD 9728		STA <DATEND+1
0747	7617 5052494E54	FCC /PRINTER /				0841	76DF 9729		STA <DATEND+2
	455220					0842	76E1 C600		LDL #11
0748	761F 00	FCB 0				0843	76E3 A700		SAVE1: STA ,X+
0749	7620 4F4E	FCC /ON/				0844	76E5 5A		DECB
0750	7622 00	FCB 0				0845	76E6 26F0		BNE SAVE1
0751	7623 4F4646	FCC /OFF/				0846	76E8 17018D		LBSR MOTON
0752	7626 00	FCB 0				0847	76EA C610		LDL #510
0753	7627 206F683F20	FCC /ok? (Y)/				0848	76ED BEFFFF		WAIT: LDX #5FFFF
	205929					0849	76F0 170105		LBSR TIMER
0754	762F 00	FCB 0				0850	76F3 5A		DECB
0755	7630 0A00	FDB \$0A00				0851	76F4 26F7		BNE WAIT
0756	7632 4572726F72	FCC /Error/				0852	76F6		; WRITE HEAD BLOCK
0757	7637 00	FCB 0				0853	76F6 C6FF		LDL #5FF
0758	7638 0A00	FDB \$0A00				0854	76F8 0D4D		BSR WRITEF
0759	763A 4164642020	FCC /Add /				0855	76FA 1F00		TFR DP,A
0760	763F 00	FCB 0				0856	76FC C630		LDL #HB
0761	7640 53756D	FCC /Sum/				0857	76FE 1F01		TFR D,X
0762	7643 00	FCB 0				0858	7700 0600		LDA #0
0763	7644 0A00	FDB \$0A00				0859	7702 C614		LDL #514
0764	7646 53756D3A20	FCC /Sum: /				0860	7704 0D4A		BSR BLKW
0765	7648 00	FCB 0				0861	7706 C6FF		LDL #5FF
0766	764C 0A00	FDB \$0A00				0862	7708 0D3D		BSR WRITEF
0767	764E 466F756E64	FCC /Found /				0863	770A		; WRITE FIRST DATA
	20					0864	770A DE23		LDL <DATH+1
0768	7654 00	FCB 0				0865	770C 334A		LEAU 10,U



## 独立モニタ アセンブル・リスト

```

0066 770E 1F60      LDR DP,A
0067 7710 C622      LDB #DATH
0068 7712 1F01      IFR D,X
0069 7714 118300FF WRDATA:CMFPU #FFF
0070 7718 220E      BHI WRCONT
0071 771A 11830000 CMFPU #0
0072 771E 2714      BEO WREND
0073 7720      : LAST DATA BLOCK
0074 7720 1F30      IFR U,D
0075 7722 8601      L0A #1
0076 7724 802A      BSR BLKW
0077 7726 200C      BRA WREND
0078 7728      : MAIN BODY
0079 7728 8601      WRCONT:L0A #1
0080 772A C6FF      LDB #*FF
0081 772C 8022      BSR BLKW
0082 772E C60A      LDB #10
0083 7730 8D15      BSR WRITEF
0084 7732 20E0      BRA WRDATA
0085 7734      : WRITE END BLOCK
0086 7734 C60A      WREND: LDB #10
0087 7736 800F      BSR WRITEF
0088 7738 86FF      LDB #*FF
0089 773A C600      LDB #0
0090 773C 8D12      BSR BLKW
0091 773E C6FF      LDB #*FF
0092 7740 8D05      BSR WRITEF
0093 7742 170139      LBSR MOTOFF
0094 7745 113F      FDB #113F      ; SW12
0095 7747      :
0096 7747      :
0097 7747 86FF      WRITEF:LDA #*FF
0098 7749 170084      LBSR WRITE
0099 774C 5A      DECB
0100 774D 26F8      BNE WRITEF
0101 774F 39      RTS
0102 7750      : CMT ONE BLOCK WRITE
0103 7750 3402      BLIW: PSHS A
0104 7752 3410      PSHS X
0105 7754 8E00FF      LDB #*FF
0106 7757 17009E      LBSR TIMER
0107 7759 3510      PULS A
0108 775C 8601      LDB #1
0109 775E 17009F      LBSR WRITE
0110 7761 862C      LDB #*3C
0111 7763 17009A      LBSR WRITE
0112 7766 3502      PULS A
0113 7768 170095      LBSR WRITE
0114 776B 9711      STA SUM
0115 776D 1F98      IFR B,A
0116 776F 17008E      LBSR WRITE
0117 7772 9B11      ADDA <SUM
0118 7774 9711      STA <SUM
0119 7776 C100      WLOOP: CMFB #0
0120 7778 271A      BEO WLOOP2
0121 777A 5A      DECB
0122 777B A680      LDA ,X*
0123 777D 170080      LBSR WRITE
0124 7780 9B11      ADDA <SUM
0125 7782 9711      STA <SUM
0126 7784 325F      LEAU -1,U
0127 7786 9C2C      CMFV DATH5
0128 7788 2602      BNE WLOOP1
0129 778A 9E13      LDB ADDRESS
0130 778C 9C15      WLOOP1:CMFV <ENDAD
0131 778E 2606      BNE WLOOP
0132 7790 9E2E      LDB <DATENA
0133 7792 20E2      BRA WLOOP
0134 7794 9611      WLOOP2:LDB <SUM
0135 7796 8D68      BSR WRITE
0136 7798 39      RTS
0137 7799      : CMT ONE BYTE READ
0138 7799 342A      READ: PSHS (Y,B
0139 779B 108E0000      CBREAD:LDY #0
0140 779F 8D44      CR1: BSR DIN
0141 77A1 26FC      BNE CR1
0142 77A3 8D46      BSR EDGE
0143 77A5 8E001E      LDB #30
0144 77A8 8D4E      BSR TIMER
0145 77AA 8D39      BSR DIN
0146 77AC 26F1      BNE CR1
0147 77AE 4F      CLRA
0148 77AF 8D3A      CR2: BSR EDGE
0149 77B1 8E0019      LDB #25
0150 77B4 8D42      BSR TIMER
0151 77B6 8D20      BSR DIN
0152 77B8 58      LSLB
0153 77B9 46      RORA
0154 77BA 3121      LEAY 1,Y
0155 77BC 108C      FDB #108C
0156 77BE 0008      FDB #0008
0157 77C0 2706      BEQ CRSTOP
0158 77C2 8D21      CR3: BSR DIN
0159 77C4 26FC      BNE CR3
0160 77C6 20E7      BRA CR2
0161 77C8 108E0000      CRSTOP:LDY #0
0162 77CC 8D17      CR4: BSR DIN
0163 77CE 26FC      BNE CR4
0164 77D0 8D19      BSR EDGE
0165 77D2 8E0018      LDB #24
0166 77D5 8D21      BSR TIMER
0167 77D7 8D0C      BSR DIN
0168 77D9 27C0      BEQ CBREAD
0169 77DB 3121      LEAY 1,Y
0170 77DD 108C      FDB #108C
0171 77DF 26E9      BNE CR4
0172 77E1 3584      PULS X,Y,B,PC
0173 77E3 F6FD02      LDB #FD02
0174 77E5 C488      ANDB #000
0175 77EA 39      RTS
0176 77EB 8DF8      BSR DIN
0177 77ED C180      CMFB #000
0178 77EF 26FA      BNE EDGE
0179 77F1 8DF2      BSR DIN
0180 77F3 C180      CMFB #000
0181 77F5 26F4      BNE EDGE
0182 77F7 39      RTS
0183 77F8 301F      TIMER: LEAX -1,X
0184 77FA 8C0000      CMFV #0
0185 77FD 26F9      BNE TIMER
0186 77FF 39      RTS
0187 7800      : CMT ONE BYTE WRITE
0188 7800 3436      WRITE: PSHS Y,X,B,A
0189 7802 8E0024      LDB #36
0190 7805 8DF1      BSR TIMER
0191 7807 8D63      CBOUT: BSR ZERO
0192 7809 8E0006      LDB #6
0193 780C 8DEA      BSR TIMER
0194 780E 8D62      BSR ONE
0195 7810 8E0012      LDB #18
0196 7813 8DE3      BSR TIMER
0197 7815 8D55      BSR ZERO
0198 7817 8E0012      LDB #18
0199 781A 8DDC      BSR TIMER
0200 781C 108E0000      LDY #0
0201 7820 8D50      BSR ONE
0202 7822 44      LSRA
0203 7823 250E      CO0: BCS CO1
0204 7825 8E0011      LDB #17
0205 7828 8DCE      BSR TIMER
0206 782A 8D40      BSR ZERO
0207 782C 8E0011      LDB #17
0208 782F 8DC7      BSR TIMER
0209 7831 200E      BRA NEXT
0210 7833 8E0024      CO1: LDB #36
0211 7836 8DCC      BSR TIMER
0212 7838 8D32      BSR ZERO
0213 783A 8E0023      LDB #35
0214 783D 8D59      BSR TIMER
0215 783F 2000      BRA NEXT
0216 7841 3121      NEXT: LEAY 1,Y
0217 7843 108C      FDB #108C
0218 7845 8D08      FDB #0008
0219 7847 26D7      BNE W00
0220 7849 108E0000      LDY #0
0221 784B 8D23      W02: BSR ONE
0222 784D 8E0024      LDB #36
0223 784F 8D4A      BSR TIMER
0224 7852 8D16      BSR ZERO
0225 7854 8E000C      LDB #12
0226 7857 8D9D      BSR TIMER
0227 7859 3121      LEAY 1,Y
0228 785B 108C      FDB #108C
0229 785F 8D02      FDB #0002
0230 7861 2707      BEQ BYTEND
0231 7863 8E0016      LDB #22
0232 7866 8D9E      BSR TIMER
0233 7868 20E3      BRA W02
0234 786A 3586      BYTEND:PULS X,Y,B,A,PC
0235 786C C6C2      ZERO: LDB #*C2
0236 786E F7FD00      STB #FD00
0237 7871 39      RTS
0238 7872 C6C3      ONE: LDB #*C3
0239 7874 F7FD00      STB #FD00
0240 7877 39      RTS
0241 7878 C6C2      MOTON: LDB #*C2
0242 787A F7FD00      STB #FD00
0243 787D 39      RTS
0244 787E C6C0      MOTOFF:LDB #*C0
0245 7880 F7FD00      STB #FD00
0246 7883 39      RTS
0247 7884      : CMT ONE BLOCK READ
0248 7884      BLKR: LDY #0
0249 7886 108E0000      BR0: LBSR READ
0250 7888 17FF0E      CMFV #*FF
0251 788B 81FF      BNE BR1
0252 788D 2684      LEAY 1,Y
0253 788F 3121      BRA BR0
0254 7891 20F5      BR1: CMFV #*01
0255 7893 8101      BNE BLKR
0256 7895 26E0      FDB #108C
0257 7897 108C      FDB #0008
0258 7899 23E7      BLS BLKR
0259 789B 813C      LBSR READ
0260 789D 17FEF9      CMFV #*3C
0261 78A0 813C      BNE BLKR
0262 78A2 26E0      LBSR READ
0263 78A4 17FEF2      STA <TYPE
0264 78A7 9710      STA <SUM
0265 78A9 9711      CMFV #0
0266 78AB 8100
```

```

1074 788C 9B11      ADDA <SUM
1075 788E 9711      STA <SUM
1076 78C0 C100      BRLOOP:CMPS #0
1077 78C2 2729      BEQ BRL0
1078 78C4 5A        DECB
1079 78C5 17FE01    LBSR READ
1080 78C8 3404      PSHS #
1081 78CA D617      LDB <CPF
1082 78CC 2706      BEQ BRL4
1083 78CE D619      LDB <VF
1084 78D0 2602      BNE BRL4
1085 78D2 A700      STA ,X+
1086 78D4 3504      BRL4: PULS #
1087 78D6 9B11      ADDA <SUM
1088 78D8 9711      STA <SUM
1089 78DA 3341      LEAU 1,U
1090 78DC 11830000   CMPU #0
1091 78E0 2602      BNE BRL1
1092 78E2 9E13      LDX <ADDRES
1093 78E4 119323    BRL1: CMPU <DATA+1
1094 78E7 26D7      BNE BRLOOP
1095 78E9 9E2E      LDX <DATENA
1096 78EB 20D3      BRL2: BRA BRLOOP
1097 78ED 17FEA9    BRL0: LBSR READ
1098 78F0 9111      CMPS <SUM
1099 78F2 2601      BNE ERROR
1100 78F4 39         RTS
1101 78F5 17FFB6    ERROR: LBSR MOTOFF
1102 78F8 C640      LDB #ER
1103 78FA 17FCDF    LBSR DPCX
1104 78FD 113F      FDB #113F
1105 78FF          ;
1106 78FF          ; LOAD & VERIFY
1107 78FF          ;
1108 78FF 17F8C0    LOAD: LBSR PRINT
1109 7902 17F89D    LDB: LBSR INKEY
1110 7905 C600      LDB #0
1111 7907 B144      CMPS #44
1112 7909 2706      BEQ LD1
1113 790B 813F      CMPS #3F
1114 790D 26F3      BNE LD0
1115 790F C601      LDB #1
1116 7911 0719      LD1: STB <VF
1117 7913 17F8B9    LDB: LBSR PRINT
1118 7916 1F00      TFR DP,A
1119 7918 C627      LDB #DATEND
1120 791A D02E      STD <DATENA
1121 791C 8D7D      BSR NAMEIN
1122 791E D619      LDB <VF
1123 7920 2605      BNE LD2
1124 7922 17F9A4    LBSR GETN2B
1125 7925 D013      STD <ADDRES
1126 7927 0F10      LD2: CLR <FOUND
1127 7929 17FF4C    LDB: LBSR MOTON
1128 792C 1F00      LOAD1: TFR DP,A
1129 792E C630      LDB #HB
1130 7930 D02C      STD <HBAD
1131 7932 CE0000     LDU #0
1132 7935 0F23      STU <DATA+1
1133 7937 0F17      CLR <CPF
1134 7939 17FF4B    LOAD2: LBSR BLR
1135 793C 9610      LDA ,TYPE
1136 793E 8100      CMPS #0
1137 7940 264C      BNE LOAD3
1138 7942          ; When TYPE = HEAD
1139 7942 3406      PSHS A,B
1140 7944 17FF37    LBSR MOTOFF
1141 7947 1F00      TFR DP,A
1142 7949 C630      LDB #HB
1143 794B 1F01      TFR D,I
1144 794D C61A      LDB #NAME
1145 794F 1F02      TFR D,Y
1146 7951 C600      LDB #0
1147 7953 A600      NAMC0: LDA ,X+
1148 7955 A1A0      CMPS ,Y+
1149 7957 260F      BNE NAMC1
1150 7959 5A        DECB
1151 795A 26F7      BNE NAMC0
1152 795C C65C      LDB #LD
1153 795E 9630      LDA HB+0
1154 7960 8102      CMPS #2
1155 7962 2691      BNE ERROR
1156 7964 B601      LDA #1
1157 7966 2004      BRA NAMC2

```

; NOT THIS FILE

; VERIFY

; SW13

; Machine Code PGM ?

; CPF FOUND set

```

1158 7968 8600      NAMC1: LDA #0
1159 796A C665      LDB #SK
1160 796C 9717      NAMC2: STA <CPF
1161 796E 9718      STA <FOUND
1162 7970 17FC69    LBSR DPCX
1163 7973 1F00      TFR DP,A
1164 7975 C630      LDB #HB
1165 7977 1F01      TFR D,X
1166 7979 C600      LDB #0
1167 797B 17F9A3    LBSR DPC
1168 797E 1F00      TFR DP,A
1169 7980 C622      LDB #DATAH
1170 7982 1F01      TFR D,X
1171 7984 CEFFF8    LDU #5
1172 7987 17FEEE    LBSR MOTON
1173 798A 3506      PULS A,B
1174 798C 20A8      BRA LOAD2
1175 798E 81FF      LOAD3: CMPS #FFF
1176 7990 26A7      BNE LOAD2
1177 7992 9610      LDA <FOUND
1178 7994 2796      BEQ LOAD1
1179 7996 17FEE5    LBSR MOTOFF
1180 7999 113F      FDB #113F
1181 799B 17F7E4    NAMEIN: LBSR SPC
1182 799E 1F00      TFR DP,A
1183 79A0 C61A      LDB #NAME
1184 79A2 1F01      TFR D,X
1185 79A4 C600      LDB #0
1186 79A6 17F7F9    NAM0: LBSR INKEY
1187 79A9 810D      CMPS #D
1188 79AB 2711      BEQ NAM1
1189 79AD 8108      CMPS #B
1190 79AF 2715      BEQ NAM2
1191 79B1 8120      CMPS #20
1192 79B3 25F1      BCS NAM0
1193 79B5 17F817    LBSR PRINT
1194 79B8 A700      STA ,X+
1195 79BA 5A        DECB
1196 79BC 26E9      BNE NAM0
1197 79BD 39         RTS
1198 79BE 8620      NAM1: LDA #20
1199 79C0 A700      STA ,X+
1200 79C2 5A        DECB
1201 79C3 26F9      BNE NAM1
1202 79C5 39         RTS
1203 79C6 C100      NAM2: CMPS #0
1204 79C8 270C      BEQ NAM0
1205 79CA 8620      LDA #20
1206 79CC 301F      LEAX -1,X
1207 79CE A704      STA ,X
1208 79D0 3404      PSHS #
1209 79D2 C66E      LDB #BS
1210 79D4 17FC05    LBSR DPCX
1211 79D7 3504      PULS #
1212 79D9 5C        INCB
1213 79DA 20CA      BRA NAM0
1214 79DC 5C        ORG $79E0
1215 79DE          ; LINE INPUT
1216 79E0 8E042D    LDX #8042D
1217 79E3 17F7A4    L10: LBSR INPUT
1218 79E6 A700      STA ,X+
1219 79E8 810D      CMPS #D
1220 79EA 26F7      BNE L10
1221 79EC 860A      LDA #A
1222 79EE 17F70E    LBSR PRINT
1223 79F1 8E042C    LDX #8042C
1224 79F4 39         RTS
1225 79F6          ORG $79FB
1226 79F8 960D      TIMIR: LDA #AST
1227 79FD 38        RTI
1228 79FE          ;
1229 79FE          ; END
1230 79FE          ;
1231 79FE          ; NM0N Version 1.0

```



# 音声モニタ

ダンフ・リストを読み上げるモニタ

会計検査院マイコンクラブ 滝沢吉人

音声モニタはMZなどで既に発表されているので、どういうものか知っている人も多いと思います。これはマシン語入力時にリアルタイムでその値を音声で出力したり、入力した内容を読みあげたりするものです。したがって、本に掲載されているマシン語を入力した後は、本を見ながらFMの読み上げてくれる声を聞けば、入力ミスを見つけることができるわけです。

## 使い方

原理はオーディオ・カセット出力アドレス(\$FD00)に音声データを書き込んで声を出しているため、本体から声は出ないのでもカセット用のコネクタのマイク端子を接続し録音状態にしてください。

リスト1、2のプログラムを入力してRUNしてください。最初に「スタート アドレス?」と聞いてくるので、入力したい番地または読み上げて欲しい番地を入力してくださ

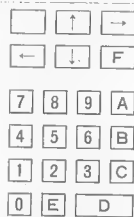
い。番地を入力するとそこから256バイト分表示し、入力待ちになります。後は16進でデータを打ち込んでください。この際、FM-8ではテン・キー部が図1のようになります。なお、PF6～PF9のキーは次のように定義されています。

PF6: チェック・サムを出力  
PF7: 次に進む  
PF8: END  
PF9: 画面を読む

## 最後に

大事なことを書き忘れました。FM-7でこのプログラムを使うときはディップ・スイッチの4をオンにして低速モードにしてください(高速で実行すると何を言っているのかわからなくなるので)。

図1 テン・キーの配列



### リスト1 音声モニタBASIC プログラム・リスト

```
20 CLEAR 25, &H17FF:DEFINT A,B,C,D,T,S,O,P,X,Y,Z:DIM D%(17):DIM S%(16):DEF USR=&
H1830:DEF FNCH$(E,P!)=RIGHT$("0000"+HEX$(P!),E):ONKEY(6)GOSUB200:ONKEY(7)GOSUB230
:ONKEY(8)GOSUB220:ONKEY(9)GOSUB250:WIDTH80,20:KEY6,"チェックサム:KEY7,"ツキニスム":
KEY8,"END"
30 KEY9,"カメンヲヨム":KEY10,"スタートアドレス:KEY1," ":KEY2," ":KEY3," ":KEY4," ":CON
SOLE 0,18,1,1: B=PEEK(&H1830):IF B = &H34 THEN 40ELSE LOADM"VM"
40 PRINT"FM-7 FMB オンセイ モニタ"
50 PRINT:PRINT INPUT I17 &H2E00 - &H7FFF ( &H6FFF DISK MODE )"
60 INPUT"スタート アドレス":S$:S=VAL("&H"+S$):KEY5,"&H "+FNCH$(4,S):IFS$=""THEN GOTO60EL
SE IFS < &H2E00 THEN 50
70 CLS:PRINT"Addr +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Su
m":LOCATE9,17:PRINT"Sum":
80 LINE(56,8)-(568,8),OR:LINE(56,0)-(56,179),OR:LINE(56,169)-(568,169),OR:3:LINE
(568,0)-(568,179),OR:LINE(508,0)-(508,179),OR:LINE(112,0)-(112,179),OR:LINE
(0,179)-(631,179),OR,3
90 KEY(6)ON:KEY(7)ON:KEY(9)ON:KEY(8)ON
100 FOR A=S TO &H7FFF STEP 256:FOR X=0 TO 15:LOCATE9,X+1:PRINTFNCH$(4,X*16+A):FO
R Y=0 TO 15:D%(Y)=CVI(FNCH$(2,PEEK(A+X*16*Y))):NEXT Y:PUT$(15+X*3,1)-(15+X*3 +1,
16),D%,4:NEXT X:X=15:Y=1
110 IF X < 15 THEN X=61 ELSE IF X>61 THEN X=15:Y=Y+1
120 IF Y < 1 THEN Y=16 ELSE IF Y>16 THEN Y=1
130 I$=INKEY$:B=SCREEN(X,Y):LOCATEX,Y:PRINTCHR$(&H87):LOCATEX,Y:PRINTCHR$(B):IF
X MOD 3=0 THEN C=2:D=1 ELSE C=1:D=2
140 IF I$=CHR$(127) THEN X=X+D:GOTO110ELSE IF I$=CHR$(&H1D) THEN X=X-C:GOTO110ELSE IF I$=
CHR$(&H1E) THEN Y=Y-1:GOTO110ELSE IF I$=CHR$(&H1F) THEN Y=Y+1:GOTO110ELSE IF I$="", "THEN I
$="A"ELSE IF I$="+ " THEN I$="B"ELSE IF I$="- " THEN I$="C"ELSE IF I$="," THEN I$="E"
150 IF I$=CHR$(&H0D) THEN I$="D"ELSE IF I$=CHR$(&H1C) THEN I$="F"
160 IF "0" > I$ THEN 110ELSE IF I$ > "F" THEN 110
170 LOCATEX,Y:PRINT I$:X=X+D:F$=USR(I$)
180 GOTO 110
```



## リスト1 音声モニタBASIC プログラム・リスト

```

190 NEXT:END
200 FOR O=1 TO 16:S%(O)=0:NEXT:FOR P=15 TO 60 STEP 3:GET@(P,1)-(P+1,16),D%:T=0:FOR
R O=1 TO 16:Z=VAL("&H"+MKI$(D%(O-1))):S%(O)=S%(O)+Z:T=T+Z:NEXT O:LOCATE P,17:PRINT
FNCH$(2,T):NEXT P:T!=0:FOR O=1 TO 16:LOCATE 65,O:PRINTFNCH$(2,S%(O)):T!=T!+S%(
O):NEXT O
210 LOCATE 65,17:PRINT FNCH$(2,T!):RETURN
220 GOSUB 240:END
230 GOSUB 240:RETURN 190
240 S=0:FOR O=15 TO 60 STEP 3:GET@(O,1)-(O+1,16),D%:FOR P=0 TO 15:POKE A+S+P*16,
VAL("&H"+MKI$(D%(P))):NEXT S=S+1:NEXT:RETURN
250 FOR P=1 TO 16
260 FOR O=15 TO 60 STEP 3
270 GET@(O*8-1,10*P-1)-(O*8+16,P*10+8),D%,G,4:PUT@(O*8-1,10*P-1)-(O*8+16,P*10+8
),D%,PRESET,4:PUT@(O*8-1,10*P-1)-(O*8+16,P*10+8),D%,NOT,4
280 F$=USR(CHR$(SCREEN(O,P))) :F$=USR(CHR$(SCREEN(O+1,P)))
290 GET@(O*8-1,10*P-1)-(O*8+16,P*10+8),D%,G,4:PUT@(O*8-1,10*P-1)-(O*8+16,P*10+8
),D%,PRESET,4:PUT@(O*8-1,10*P-1)-(O*8+16,P*10+8),D%,NOT,4
300 NEXT O,P:RETURN

```

## リスト2 音声モニタ ダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1800	18	62	19	E0	18	40	1C	50	1D	F0	1F	60	20	F0	21	F0	E7
1810	23	70	25	00	26	80	89	AA	AB	BC	DE	DD	EF	FF	67	87	8F
1820	00	00	00	26	80	27	80	28	5A	2A	A9	2B	A9	2C	AC	2D	FF
1830	34	37	E6	9B	01	00	30	5B	86	18	1F	01	31	02	AE	84	55
1840	A6	80	C6	08	48	34	02	04	86	01	20	01	4F	B7	FD	46	
1850	00	86	09	4A	26	FD	35	02	5A	26	E9	AC	A4	25	E1	35	27
1860	37	3C	7F	BF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	4B
1870	50	00	0F	F7	FB	FF	55	F5	75	7F	EA	F7	87	EA	F7	4E	
1880	B0	52	A8	57	EF	95	6F	6E	84	64	BA	B6	10	55	53	D5	9A
1890	DD	A9	2A	AA	8D	57	84	AF	4A	F5	EB	2A	76	ED	76	AB	A9
18A0	55	4C	1F	FA	90	85	FF	86	5C	CB	F8	C0	47	7F	EB	0F	
18B0	00	01	FF	FF	00	F8	80	0F	FF	F0	00	3F	FC	00	07	FF	B6
18C0	FF	20	00	3F	FF	C0	0F	EC	00	1F	FF	80	7C	C0	0F	FF	00
18D0	F0	5F	00	01	FF	FC	03	E6	00	7F	06	F0	00	1F	FD	8C	
18E0	80	FC	40	0F	FE	C0	7E	00	0F	FF	E0	3F	00	03	FE	E0	0E
18F0	3F	80	01	FF	F0	1F	80	00	7F	FB	07	C0	00	7F	FC	0F	96
Sum:	45	8B	DB	44	F9	63	3A	C6	A9	46	9C	0C	C9	CA	D5	04	4E

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1900	E0	00	3F	FE	03	C0	00	3F	FC	00	F8	00	CF	FE	E0	06	E7
1910	E4	67	8F	FC	FF	FC	88	90	FF	FF	E7	00	3F	FE	00	E9	
1920	7E	00	03	FF	00	7F	00	00	FF	00	F0	80	00	7F	F8	07	EB
1930	E0	00	3F	FC	03	F8	00	0F	00	FC	00	03	FE	C0	3F	123	
1940	80	00	FF	E0	7F	00	3F	F8	07	FE	00	07	FF	81	FF	12A	
1950	E0	00	FF	E0	7F	00	3F	F8	07	FE	00	07	FF	81	FF	12A	
1960	E0	00	FF	E0	7F	00	3F	F8	07	FE	00	07	FF	81	FF	12A	
1970	FF	00	03	FF	B0	1F	F8	00	1F	FE	00	FF	E0	00	FF	03	133
1980	03	FF	00	03	FF	C0	0F	FC	00	0F	FF	E0	00	7F	E0	00	7F
1990	FC	00	FF	80	01	FF	F0	01	FE	00	07	FF	E0	03	FE	00	181
19A0	07	FF	E0	03	FC	00	03	FF	E0	03	F8	00	1F	FF	E0	07	1C7
19B0	F0	00	07	FF	C0	0F	E4	00	0F	FF	C0	F8	00	00	7F	FF	1D0
19C0	B0	1F	88	00	FF	FF	00	7F	20	03	FF	FC	00	F0	00	DF	9A
19D0	FF	E0	03	FC	03	1F	FF	C8	0F	98	00	FF	00	1F	31	83	
19E0	01	DF	F0	00	F9	84	8F	FF	EE	06	E4	34	7F	7F	33	38	ED
19F0	C3	23	FF	FB	01	26	13	1B	BF	59	98	C0	63	3F	FF	40	1E6
Sum:	CA	66	7D	17	C7	B6	07	00	D4	6E	AE	3C	EB	52	35	4A	30

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1A00	04	36	33	7F	F6	30	80	43	7F	FD	46	08	0C	0F	FF	FC	1D0
1A10	B0	80	C0	00	FF	B9	89	84	06	FF	B8	00	00	7F	FF	FC	1FA
1A20	F7	04	80	08	FF	FF	F0	24	00	00	1F	FF	B1	20	00	FF	145
1A30	FF	F4	00	00	1F	FF	FF	E0	02	27	FF	FF	FB	EB	6C	89	1F2
1A40	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	1B8
1A50	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	1F0
1A60	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	1F0
1A70	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	1F0
1A80	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	1F0
1A90	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	1F0
1AA0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	1F0
1AB0	B5	35	ED	75	7B	F5	D5	6B	AB	AA	F7	1A	8D	6C	ED	64	1DC
1AC0	DA	B7	C2	AF	6B	DD	AF	6A	EA	ED	7E	AD	6A	BA	CB	08	10B
1AD0	AD	75	8A	DB	39	A5	56	64	D5	83	53	D7	7B	6B	AE	AA	115
1AE0	CB	FF	FC	E2	19	DD	FF	FB	10	04	7F	FF	FA	00	80	07	137
1AF0	FF	FE	40	40	01	BF	FF	E0	00	49	61	FF	FF	90	01	21	176
Sum:	79	E2	96	7D	89	42	9B	23	5A	BF	57	9C	B1	63	4E	B3	1EB

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1B00	07	FF	FE	14	09	7F	FF	FC	90	40	29	FF	FF	90	01	E0	
1B10	00	27	FF	FE	00	0C	00	1F	FF	FF	10	00	1F	FF	FF	77	
1B20	F8	00	00	3F	FF	FF	FC	00	00	1F	FF	FF	B0	00	FF	1D0	
1B30	FF	FE	00	00	07	FF	FF	F8	00	00	1F	FF	FF	C0	00	00	1D7
1B40	FF	FF	FF	FF	00	03	FF	FF	FC	00	01	FF	FF	FE	E0	00	1F7
1B50	00	7F	FF	FF	80	00	07	FF	FF	F0	00	00	1F	FF	FF	80	1BF
1B60	00	00	7F	FF	FC	00	00	07	FF	FF	F0	00	00	3F	FF	F2	19F
1B70	F8	00	00	7F	FF	E0	08	00	1F	FF	FE	00	00	00	FF	FF	178
1B80	FC	02	00	03	FF	FE	00	12	00	7F	FF	FE	01	20	01	FF	1AD
1B90	FF	E0	12	00	0F	FE	FE	44	90	03	FF	FF	F0	49	00	3F	14A
1BA0	FF	FF	FF	24	90	00	7F	FF	F0	04	80	1B	FF	FF	99	24	17A
1BB0	1F	FF	FC	09	20	01	FF	FF	C2	49	00	1F	FF	FE	02	48	1B3
1BC0	00	3F	FF	EC	03	20	03	FF	FF	B0	09	00	3F	FF	FE	80	192
1BD0	40	00	7F	FF	C0	01	20	01	FF	FF	E0	08	00	1F	FE	EC	190
1BE0	00	26	02	7F	FE	F4	80	90	B1	FF	FF	B0	06	40	2F	FE	1AD

1BF0	FB	00	09	08	0F	FF	FB	00	24	23	7F	FF	FE	E0	00	B0	41	181
Sum:	4C	E7	F5	DC	89	87	22	F0	D1	86	CB	28	B6	79	3E	A2	17F	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
1C00	FF	FE	C0	01	00	07	FF	FD	B0	02	00	07	FF	FB	00	26	16A	
1C10	20	1F	FF	FE	B0	02	00	0F	FF	EC	00	01	01	3F	FF	FE	16A	
1C20	00	00	00	1F	FF	FE	00	08	00	03	FF	FF	B0	00	AA	76	1C5	
1C30	FF	FF	FF	B0	40	15	AF	5F	FF	FF	D5	05	49	6F	7F	FD	1C4	
1C40	B8	D5	F5	79	5F	AE	AA	AF	4D	7B	6F	6E	BA	AF	AD	6A	189	
1C50	D7	B5	D5	5E	AB	6B	DE	AD	D3	DD	EA	FB	5F	EB	FD	56	192	
1C60	FA	B5	7F	D7	FF	FD	5F	DE	BD	77	FF	FF	FF	FF	FF	FF	142	
1C70	7F	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	1C4	
1C80	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	1C4	
1C90	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	1C4	
1CA0	FF	FF	7D	6F	FF	FF	FF	B7	FF	FF	FF	37	F7	FE	DF	51		
1CB0	FF	FF	FF	FF	55	ED	AF	AE	AF	68	D5	7B	EA	FF	5E	2F	79	17F
1CC0	EB	AD	FF	55	ED	AF	AE	AF	68	D5	7B	EA	FF	5E	2F	79	17F	
1CD0	A7	EF	EF	F1	FD	55	68	DA	74	5B	DD	CF	AF	EE	85	7B	11C	
1CE0	C2	E7	FC	3F	DC	70	01	EB	F5	FC	F7	FE	00	0F	FF	87	1A3	
1CF0	F0	00	7F	E0	FF	FC	80	0F	FC	07	FE	FE	00	7F	80	7F	199	
Sum:	65	D9	E2	D2	59	56	BA	ED	EC	4F	31	78	34	4A	6D	33	14A	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1D00	89	F0															

## リスト 2 音声モニタ ダンプ・リスト

[illegible][illegible]

## リスト2 音声モニタ ダンプ・リスト

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
2A00 FF FC C4 66 40 01 FF FF 20 43 61 05 FF FE E0 73 :7D
2A10 20 04 DF FF 60 31 20 03 FF FF E0 61 21 00 FF FF :14
2A20 F0 31 30 00 EF FF 90 18 90 04 FF FF F0 08 CB 00 :39
2A30 7F FF FB 0C 64 00 1F FF E4 26 33 18 0F FF FE 01 :66
2A40 19 00 27 FF FB 00 BC 80 13 FF FF B0 46 62 09 FF :87
2A50 FE E2 73 01 04 FF FF BB 00 80 43 FF FF F0 00 10 :9F
2A60 19 FF FF FB 00 03 3F FF FF FF FF FF FF FF FF :48
2A70 FF FF FF FF FF FF FF FF FF FF FF FF FF FF :F0
2A80 FF FF FF FF FF FF FF FF FF FF FF FF FF FF :F0
2A90 FF FF FF FF FF FF FF FF FF FF FF FF FF FF :F0
2AA0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF :F0
2AB0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF :F0
2AC0 FF FE 7F 3B EF E0 E6 3E E6 3E E7 6C 27 DC E3 1C :20
2AD0 E7 39 CE 73 0C C7 39 C6 FF FF FF FF FF FF FF :F0
2AE0 10 20 00 FF FF DB 8F 99 08 37 7F FC CF 11 18 67 :47
2AF0 FF FC CF 13 1B 7F FF F0 07 93 08 6F FF FB 0F 13 :8D
Sum: AE 5F 7B 21 FF 2C 40 AB 94 EC 1A BD 54 35 B1 05 :22

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
2B00 10 6F FF F9 99 99 00 37 FF F0 0F 99 08 37 FF FC :B1
2B10 46 CC B4 1B FF FC 46 E6 C2 1D FF FE 43 73 63 1C :E9
2B20 DF FF 20 39 B0 8A EF FF 20 1C 4C 66 3B FF CB 0E :5D
2B30 26 31 5D FF F2 13 8D BC C3 7B 8B BC E3 63 30 DE :A7
2B40 FF 33 6F 66 63 DB DF E4 27 CD BC C3 7B F1 80 DC :E3
2B50 9B CF 67 7F 99 9B 26 33 59 DD C4 46 79 91 98 6F :2B
2B60 66 32 1B C9 8C 07 77 71 99 8D E6 C6 61 B9 9C C6 :05
2B70 43 79 99 9B EF E7 73 01 2D 7E 0D 40 3F FF FC 00 :1C
2B80 3F 64 66 DB FB BC 80 43 67 7F 91 20 1F FF FC 00 :0F
2B90 37 E6 CE EF 7F 32 9B 0C 9E DF FF 09 40 1F FF FC :0E
2BA0 00 40 F3 67 33 9B DD 8C DB DF E6 66 33 BB DB C8 :54
2BB0 9F CD 94 CF 6F F6 40 0F 9B 31 BD 9C CE C6 EC 9B :C3
2BC0 19 ED EE E6 66 CD 9B DF 6F F6 22 39 93 99 EC EE :4A
2BD0 E6 77 36 83 3D BF DC CC CD 80 47 6F FE 33 72 40 :70
2BE0 01 BF FD 89 99 B0 09 FF FF F4 8C DB 00 76 FF 6D :D0
2BF0 32 64 00 3B 7F BC 9B 32 00 1D FF DB 44 99 00 06 :B3
Sum: E2 03 36 BF 8B CD FE F7 9D 4E 2F 1E 32 C0 0B 15 :3E

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
2C00 FF FF E6 64 80 07 7F F7 D9 02 60 03 BF FB D9 83 :B9
2C10 20 00 DF FF EC D9 B0 00 EF FD FE C0 9B 00 7F FF :33

```

```

2C20 FF 22 6C 00 3B FF 89 99 32 00 1F FF FF 60 4C B0 :94
2C30 03 7F FF F0 13 20 01 FF FE DC 4E 4C 02 7F FF 37 :CF
2C40 33 13 00 0F FF EE CC E6 C2 67 7F FF 99 99 30 08 :05
2C50 DD FD B2 26 66 22 3B FF DB 99 CC BC CF 6F 77 73 :65
2C60 36 CB CC D6 F7 66 67 EC BC CF 6F 76 E6 1B 64 66 :5B
2C70 5B 7B B4 9C DB 63 76 9B D9 89 06 E6 CE CD 77 33 :0B
2C80 31 1B 79 33 33 DB DD BC B6 66 EC CC CD FF FF 30 :0E
2C90 00 01 FF F6 42 6F FF FB B0 00 37 FF F9 11 19 FF :79
2CA0 FE E0 00 03 FF F7 73 3E 64 66 5B D9 BD 84 43 7F :B9
2CB0 F7 90 00 4F FF FF 0C FF F9 81 99 6F 77 30 80 CF :B7
2CC0 FF F0 00 0F FF FE 32 1B FF 26 33 6F EF 70 10 FF :7D
2CD0 FF 00 00 ED FF FC 9B C3 BD BC 01 3F FF 40 1F FC :55
2CE0 90 83 FF FB 02 3F FF DB C4 26 DD EC 0B DF FF F0 :AB
2CF0 01 FF FB 00 7F FF B0 0F FF C0 07 FF E0 01 FF F0 :9A

```

Sum: 77 E1 D1 69 E3 50 71 84 D9 7B BA A1 44 1E 2D A5 :9A

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
2D00 03 FF FB 00 FF F0 01 FF FB 00 FF E0 03 FF F0 01 :B3
2D10 FF C0 0F FF E0 03 FF 00 3F FF 00 0F FE 00 FF FC :75
2D20 00 7F F0 07 FF E0 03 FF 00 3F FF 00 1F FB 03 FF :AD
2D30 D0 01 FF B0 3F FE 00 3F FC 01 FF F0 01 FF C0 3F :B7
2D40 FE 00 3F FC 03 FF C0 01 FF 80 7F FB 00 7F F0 3F :A0
2D50 FC 00 1F FC 1E EE 00 07 FE 0F FB 40 0F FF 01 C1 :40
2D60 00 0F FC 03 E0 00 0F FE B0 C3 80 07 FF 10 2F C0 :C3
2D70 07 CF 61 FF F4 00 07 FF E2 00 0B 1F FB C3 BC 00 :80
2D80 7F 8F 0B E0 7F 93 5B 03 CB EC F0 0F FE 0F 1D E1 :8F
2D90 FE 0F B1 F9 E3 3E C0 FE 1C 3C FC 61 C0 FF 07 F8 :D9
2DA0 F0 BC 7B 03 FF FC 17 AB 0F 0F C9 F0 E0 3F 00 :66
2DB0 3F 9B 0F B0 7F 83 FE 70 B1 FB 07 E4 7C 3F 3E :5E
2DC0 07 7F B0 1F DB 80 DB 07 FE C0 FE 0F E0 3E 03 :E0
2DD0 1F 87 FF 07 C2 03 C8 3F FF F0 3F 80 3F FE 00 :62
2DE0 C0 03 E0 3F FC 7E 00 7C 00 7F FF A0 F7 0E 00 1F :1A
2DF0 FF FF B0 07 C0 03 FF FF E0 7B 30 00 0F FF F4 01 :D1
Sum: 60 7E 2C D7 B9 3E 2D AA D2 20 5A AC DE FA 27 1D :C3

```

## RANDOM

## BOX

## FM-7

## ボーレート・アップ用プログラム

■安田弘幸

CLEAR, &H7FBF[CR]をしてからリストを打ち込んで、EX EC &H7FC0[CR]をしてください。

これで、以下のSAVE(M)、LOAD(M)etc.(は約2,600bpsになります。つまりFM-8のクロックを8MHzにしている人と同じになります。ただし、後ろのDip SWは8MHz用にしておく必要があります。

原理はいたって簡単で、まずROMの内容を裏RAMに移して8Mと4MHzの判断をするところにNOPを入れ、裏RAMのBASICをスタートさせているだけです。つまりフルRAMで走っているマシ語を使うときは注意してください。

一回EXECさせると、あとはマシ語のロードをやらない。または、EXEC &H...(他のアドレス)をやらない限り、EXEC [CR]の繰り返しで2,600ボーと1,600ボーが切り換わります。

※注 カセット・テレコは質の良いものを、そうでないとすぐにDevice I/O Errorがでます。

```

7FC0 8E 00 00 F6 FD 0F A6 B4 F7 FD 0F A7 B0 BC FC B0 :6C
7FD0 26 F1 CC 12 12 FD F2 B3 FD F2 B5 FD F3 35 FD F3 :62
7FE0 37 CC 7F EA FD 02 17 7E BF D9 F6 FD 0F CC 7F C0 :75
7FF0 FD 02 17 7E BF D9 00 00 00 00 00 00 00 00 00 :FC
sum: E8 3F 62 70 9B E7 AF B5 B3 CB BA A1 B2 BD 7B 33 :3F

```

## FM-8

## UNLIST, Protectの解き方

FM-8の“UNLIST, Protectの解き方”は、プログラムを使わなくても解けます。

UNLIST → POKE &H1E7, &HFF : POKE &H1E8, &HFF  
 Porotect → MONからM コマンドで00D1(16)に00を書き込む、とすることによってすべてOKです。(3Dファン)





## 1パス・アセンブラ,逆アセンブラを持つ

## SUPER MONITOR

■アーニー・バツネクス

## 概要

このプログラムはマシン語開発の際に必要と思われるいくつかの機能を7Kバイトのエリアに収めています。ワークもすべてこの範囲内(図1)にあるのでユーザーが意図しない限りは別エリアを壊してしまうことはありません。また、プログラムはポジション・インデペンデントで、自分自身を転送することもできます。主な機能は次のとおりです。

- ①ページ(256バイト)単位のダンフ、エディット
- ②逆アセンブラ
- ③1パス・アセンブラ
- ④トレースを含むエミュレート
- ⑤ディスクの直接管理

## コマンドの説明

まず、起動の手順を説明します。BASICのCLEARコマンドで保護エリアを設けた後、プログラムをロードします。起動は先頭番地でEXECを実行してください。コマンドは表1のとおりですが、以下に詳しく説明します。

表1 コマンド表

コマンド	説明
O	出力デバイス選択 S:画面 L:プリンタ
C	ブロック・クリア 先頭番地, 終了番地, データ
M	ブロック転送 先頭番地, 終了番地, 転送先番地
P	メモリ・ダンフ&エディット
D	逆アセンブル 単にリターン・キーを押すと継続と見なす
A	1パス・アセンブル
Q	BASICへ復帰
B	ブレーク・ポイント設定
T	トレース開始アドレス設定
G	ユーザープログラム実行
R	ディスク1セクタ・リード
W	ディスク1セクタ・ライト
H	シーク0トラック

※) C, M, P, A, Gコマンドを実行するときはこのプログラム自身を壊さないように注意してください。

## モニタ機能

## ★Oコマンド：出力デバイス選択

メモリ・ダンフ、逆アセンブル、トレース時の出力デバイスを指定します。“device”と聞いてくるので“S”または“L”で答えてください。“S”は画面。“L”はプリンタです。なお、スタート時は“S”が選択されています。

## ★Cコマンド：ブロック・クリア

指定された番地を同一のデータで埋めます。先頭番地、終了番地、データをカンマで区切って入れてください。

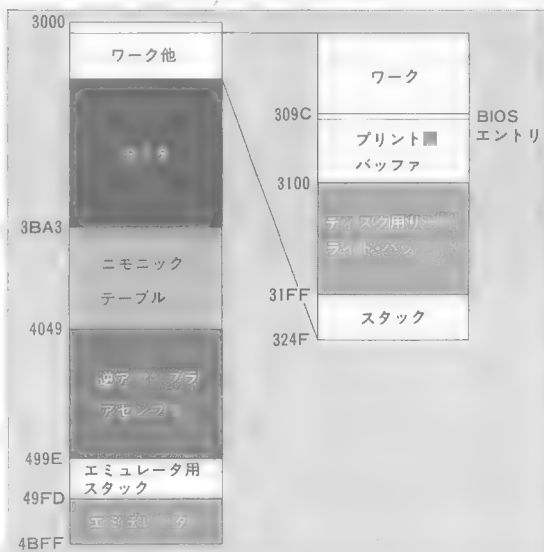
## ★Mコマンド：ブロック転送

メモリの内容を他の番地に転送します。先頭番地、終了番地、転送先番地をカンマで区切って入れてください。なお、領域の重なった転送も許されます。

## ★Pコマンド：メモリ・ダンフ&amp;エディット

チェック・サム付きでメモリ・ダンフを行います。出力デバイスに“S”を指定した場合、カーソル・エディットで書き換えることができます。このとき、チェック・サムもリアルタイムに変わります。[ESC]キー(はHEX)・[A]SCIIコードの切り換えで、リターン・キーでコマンドに戻ります。

図1 メモリ・マップ





なお、ブレーク・ポイントとトレース開始アドレスは、同時に1個しか設定できません。

### ★Gコマンド：ユーザープログラムの実行

このGコマンドはBASICのEXECコマンドやモニタのGコマンドと違い、次のメリットがあります。

- ①永久ループに落ちた場合など**STOP**キーで強制的に引き戻せる。また、その番地を知ることができる。
- ②ブレーク・ポイントを設定した場合、その時点のレジスタ値を表示する。
- ③トレース機能がある（Tコマンド参照）。

また、一度このコマンドを実行すると、どういった終わらせ方をしても設定ポイントを解除（SWIに換えて本来のコードに復帰）します。トレースは基本的にはアドレス順にポイントがシーケンシャルな動きをします。ただし、エクステンド・モードのJMP命令やブランチ（サブルーチン・コールを除く）はエミュレートして分岐します。JSR、R TSおよびTER、PULによるPCの更新には対応しません。ポイントが次のステップで待ち続ける結果となります。

## ディスク管理機能

### ★R、Wコマンド：1セクタ・リード、ライト

ユニット、クラスタ、セクタを聞いてくるのでカンマで区切って入力してください。ユニットはドライブ番号、クラスタはFATやディレクトリに書き込まれた番号に+8したものです。トラック0のセク

#### 出力例1 逆アセンブル

3500	AB	34	ADDA	-12,Y
3502	06	EC	ROR	<#EC
3504	52		FCB	#52
3505	34	06	PSHS	B,A
3507	EC	54	LDD	-12,U
3509	34	04	PSHS	B
350B	AD	C81D	JSR	#1D,U
350E	31	BC06	LEAY	#06,PC R L:3517
3511	AD	C850	JSR	#50,U
3514	16	FFC6	LBRA	L:34DD
3517	4F		CLRA	
3518	2E	4B	BGT	L:3565
351A	2E	00	BGT	L:351C
351C	74	6F70	LSR	#6F70
351F	20	74	BRA	L:3595
3521	61		FCB	#61
3522	69	6C	ROL	12,S
3524	20	64	BRA	L:35BA

タ1～8をクラスタ0のセクタ1～8とみなしてください。

ディスク用のバッファには先頭番地+\$100から\$1FFの256バイトが用意されていて、ここへ読み込んだり、ここから書き込んだりします。M、Pコマンドにより編集などを行なってください。

### ★Hコマンド：シークロトラック

ディスクが接続されていないときなどはR、Wコマンドは受け付けられませんが、このコマンドによってレディ状態にすることができます。

なお、クラスタは0～9F、セクタは1～8です。FATはクラスタ4、セクタ1でディレクトリは4、4以降、データは8、1以降です。

## 最後に

F M-7で使う場合は\$309C、9D番地をF1、7Dに書き換えてください。なお、ディスク管理機能はソフト的に破壊されたディスクを復活することも可能ですが、書き込むときは注意してください。

#### 出力例2 トレース

PC:3500	S:49FB	U:3046	Y:3036	X:4A5D	DP:00	AB:3276	CC:89
3500	AB	34	ADDA	-12,Y			
PC:3502	S:49FB	U:3046	Y:3036	X:4A5D	DP:00	AB:3276	CC:80
3502	06	EC	ROR	<#EC			
PC:3504	S:49FB	U:3046	Y:3036	X:4A5D	DP:00	AB:3276	CC:84
3504	52		FCB	#52			
PC:3505	S:49FB	U:3046	Y:3036	X:4A5D	DP:00	AB:328A	CC:89
3505	34	06	PSHS	B,A			
PC:3507	S:49FB	U:3046	Y:3036	X:4A5D	DP:00	AB:328A	CC:89
3507	EC	54	LDD	-12,U			
PC:3509	S:49FB	U:3046	Y:3036	X:4A5D	DP:00	AB:0000	CC:85
3509	34	04	PSHS	B			
PC:350B	S:49FB	U:3046	Y:3036	X:4A5D	DP:00	AB:0000	CC:85
350B	AD	C81D	JSR	#1D,U			
PC:350E	S:49FA	U:3046	Y:3036	X:4A5D	DP:00	AB:350E	CC:80
350E	31	BC06	LEAY	#06,PC R L:3517			
PC:3511	S:49FA	U:3046	Y:3517	X:4A5D	DP:00	AB:350E	CC:80
3511	AD	C850	JSR	#50,U			

## SUPER MONITOR ダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3000	17	02	4D	16	04	BA	00	00	00	00	00	00	00	00	00	00	:3A
3010	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:28
3060	16	02	13	16	02	16	16	02	29	16	02	5A	16	02	B1	16	:8B
3070	02	A3	16	02	BB	16	02	D2	16	02	EE	16	03	16	16	03	:80
3080	37	16	03	56	16	03	6B	16	03	6E	16	03	71	16	03	74	:C8
3090	16	03	84	16	03	94	16	03	F2	16	04	09	F2	DB	00	00	:42
30A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
30B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
30C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
30D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
30E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
30F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
Sum:	7C	C0	FD	9A	DA	7D	99	ED	34	9C	20	7D	6F	1C	9C	93	:D7

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3100	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3110	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3120	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3130	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3140	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3150	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3160	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3170	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3180	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3190	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
31A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
31B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
31C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
31D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00

31E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
31F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
Sum:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3200	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3210	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3220	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3230	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3240	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3250	1F	40	32	BC	FB	34	7F	30	BC	1C	BF	FF	F6	33	8D	FD	:14
3260	E5	EC	F8	01	1F	05	CC	01	0E	FD	FF	F6	32	BC	D7	35	:57
3270	7F	1F	04	32	62	39	30	BC	ED	AF	61	3B	34	10	AE	67	:AE
3280	AC	65	22	06	E6	64	E7	80	20	F6	35	10	EC	E4	32	67	:AE
3290	1F	05	34	30	10	AE	66	10	AC	6A	22	0C	AE	6A	AC	6B	:2C
32A0	22	1C	E6	80	E7	A0	20	F6	EC	6B	A3	6A	31	AB	AE	68	:94
32B0	30	01	31	21	AC	6A	23	06	E6	B2	E7	A2	20	F6	35	30	:2E
32C0	EC	E4	32	68	1F	05	34	10	AD	42	26	03	C6	14	BC	C6	:06
32D0	0E	30	C8	5B	E7	84	6F	FA	EC	E4	34	A0	A3	E4	B5	00	:7A
32E0	62	ED	04	35	06	C3	00	62	ED	02	AD	DB	56	35	10	39	:FB
32F0	34	30	6D	42	26	03	C6	14	BC	C6	0E	30	C8	5B	E7	84	:31
Sum:	30	03	06	CD	37	DD	74	C3	C9	00	15	A3	CE	43	D9	B3	:13F

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3300	CC	00	02	ED	04	CC	00	0A	ED	06	31	06	10	AF	02	AD	:3A
3310	DB	56	35	30	39	34	32	30	C8	5B	ED	06	CC	00	01	ED	:2F
3320	04	31	06	10	AF	02	C6	14	E7	84	AD	DB	56	35	32	39	:77
3330	34	30	30	C8	5B	C6	15	E7	84	31	06	10	AF	02	AD	DB	:8C
3340	56	AD	07	27	ED	A6	06	35	30	39	34	30	C8	5B	ED	06	:A2
3350	10	E7	84	31	C8	62	10	AF	02	CC	00	04	ED	04	CC	00	:30
3360	17	ED	22	AD	DB	56	35	30	39	34	30	C8	62	5F	5C	18	:C1
3370	A6	A0	A7	80	26	F9	30	C8	5B	ED	04	B6	12	A7	B4	31	:18
3380	C8	62	10	AF	02	CC	00	50	ED	06	AD	DB	56	EC	04	6F	:34

## SUPER MONITOR ダンプ・リスト

```

3390 AB 35 30 39 34 30 6D 42 26 03 C6 14 BC C6 0E 30 :EF
33A0 CB 58 E7 84 10 AF 02 5F 6D A0 27 03 5C 20 F9 4F :A6
33B0 ED 04 AD D8 56 35 30 39 34 32 30 CB 58 C6 14 E7 :E1
33C0 84 1F 89 86 11 ED 06 31 06 10 AF 02 CC 00 02 ED :69
33D0 04 AD D8 56 35 32 87 01 E5 39 1F 89 44 44 44 :D4
33E0 C4 0F C3 30 30 81 3A 25 02 8B 07 C1 3A 25 02 :57
33F0 07 39 34 30 C6 1C 20 10 34 30 C6 1D 20 0A 34 30 :B8

```

Sum: 7A 9F ED FA CF BE AB A2 B8 18 9E FE DB C6 BA 0B :10

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3400 C6 1E 20 04 34 30 C6 1F 30 C9 62 4A E7 80 4A 2A :D0
3410 FB AD CB 23 35 30 39 34 10 30 CB 58 A7 07 86 08 :01
3420 A7 84 8D 5A AD DB 56 35 10 39 34 30 30 CB 58 86 :9F
3430 0A 6D 44 27 03 4A 8D 4C 8D 3E A7 84 31 C9 00 BA :R2
3440 10 AF 02 A6 6B 01 02 24 23 A7 07 A6 67 B1 A0 24 :99
3450 1B E6 66 27 17 C1 08 22 13 6F 06 44 24 02 CB 08 :55
3460 4A 24 02 6C 06 0E 04 AD DB 56 20 04 86 FF A7 01 :F9
3470 35 30 E0 F1 32 63 1F 05 34 10 8E 10 00 30 1F 26 :42
3480 FC 35 10 39 34 10 8E A0 00 20 F2 34 30 0F 02 AD :20
3490 CB 26 86 06 AD CB 38 AD CB 35 86 07 AD CB 38 AD :B8
34A0 CB 26 35 39 34 02 6F 42 86 FD 02 44 25 06 AD :04
34B0 43 27 02 63 42 35 02 39 00 00 00 00 00 00 00 :B1
34C0 6F 43 6F 42 6F 44 AD CB 2F 31 8C 05 00 00 00 :81
34D0 11 0C 4D 4F 4E 49 54 4F 52 20 38 33 00 00 6F :42
34E0 CB 26 86 06 AD CB 38 86 2A CB 38 29 86 07 AD CB :77
34F0 38 AD CB 2C AD CB 29 81 43 26 30 31 8C 1E 17 01 :B4

```

Sum: 65 6F E6 51 43 72 3B DF 17 27 F1 23 E0 22 EF 22 :3F

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3500 AB 34 06 FC 52 34 06 EC 54 34 04 AD CB 1D 31 8C :24
3510 06 AD CB 29 86 01 6D 5D 26 01 81 4C AD CB 41 30 5C :25
3520 86 11 A0 84 AD CB 47 AD CB 46 52 AD CB 29 A6 53 AD :7B
3530 29 86 03 E6 5B 3D 1F 98 40 8B 30 AD CB 3E A6 56 :91
3540 AD CB 29 A6 57 AD CB 29 86 02 AD CB 41 A6 5C 40 :B9
3550 8B 11 AD CB 44 A6 54 AD CB 29 A6 55 AD CB 29 A6 :2C
3560 5B 8B 02 AD CB 3E A6 58 AD CB 29 A6 58 AD CB 40 8B :48
3570 35 6D 5D 26 01 4C AD CB 41 63 5D 10 26 FE A6 C6 :D2
3580 5B E6 5B C1 10 27 08 B6 01 AD CB 3E 1E FE 3A 6F :93
3590 5B 6C 5C 0C 10 26 05 6A 5C 1E FE 32 86 2F :22
35A0 AD CB 41 86 01 AD CB 47 16 FE 1F E6 4A 47 A6 A6 :A6
35B0 A4 34 04 1F 89 E0 E7 5F AD CB 3B ED 50 A6 A4 :C1
35C0 17 00 C0 A7 5B 17 00 8B A6 5B AD CB 29 A6 5B 8B :9D
35D0 05 AD CB 41 A6 54 AD CB 29 A6 55 AD CB 29 86 02 :74
35E0 AD CB 41 86 11 A0 5C AD CB 47 A6 56 AD CB 29 A6 :45
35F0 57 AD CB 29 A6 5B C6 03 3D 1F 98 40 8B 34 AD CB :27

```

Sum: 8B FD B4 17 BF 7A 7D 1B 6E DB 0B ED 74 BA F9 BA :13

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3600 31 8C 5D 17 00 A6 34 06 EC 52 34 04 AD CB 1D 31 :F0
3610 6D 44 27 18 AD CB 26 31 8C 2C AD CB 35 AD CB 2C :8F
3620 81 59 27 05 32 63 1E FE 84 AD CB 29 AD CB 4D 6D :30
3630 CB 59 27 06 31 8C 0E AD CB 50 16 FE D1 44 69 73 :DB
3640 6B 20 45 72 72 00 53 75 72 65 20 3F 20 2B 59 :2C
3650 4E 29 00 75 6E 69 74 20 63 6C 73 74 72 20 73 :63
3660 74 72 3D 00 81 52 26 04 6F 44 20 94 81 4B 26 :12
3670 31 8C 09 8D 3A 1F 98 AD CB 26 0E 63 75 6E 69 74 :95
3680 3D 00 81 51 26 08 AD CB 26 8D 0E 6B C8 17 86 3F :86
3690 AD CB 29 AD CB 26 16 FE 4A 6B DB 08 31 3F 26 05 :02
36A0 A6 4A 47 DB 08 CC 01 D7 FD FF FA 39 C6 02 8C C6 :64
36B0 00 E7 56 4F 5F ED 5D ED 52 ED 54 86 07 AD CB 38 :E2
36C0 AD CB 26 AD CB 32 6D 6C 26 45 30 C8 66 A6 80 :CB
36D0 81 3D 26 FA 31 50 A6 80 27 2F 81 20 27 25 81 :2C
36E0 27 21 81 30 25 2A 81 39 23 0A 80 07 81 3A 25 :86
36F0 81 3F 22 1C C6 10 3D 86 04 5B 69 21 69 A4 4A 26 :FA

```

Sum: AB 27 E6 C6 E4 DA E0 B7 69 80 75 9A 16 79 A3 59 :86

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3700 FB 20 D3 6A 56 31 22 20 AD CB 6D 56 26 03 FC 50 :39
3710 32 62 16 FF 79 00 00 00 00 00 00 00 00 00 00 :22
3720 AD CB 26 31 8C 0A 86 05 AD CB 38 AD CB 35 20 3F :A3
3730 61 64 64 72 3A 20 2B 30 2B 31 20 2B 32 20 2B :94
3740 33 20 2B 34 20 2B 35 20 2B 36 20 2B 37 20 2B :89
3750 20 2B 39 20 2B 41 20 2B 42 20 2B 43 20 2B 44 :D0
3760 2B 45 20 2B 46 20 2B 47 20 2B 48 20 2B 49 20 :22
3770 CB 0C C6 21 6F 80 5A 2B 6F 6F 5A 86 05 AD CB 38 :79
3780 30 CB 62 A6 C4 AD CB 38 ED 81 A6 5A AD CB 38 E7 :DA
3790 0C C6 30 E7 80 CC 3A 20 E6 81 AD CB 23 6F 59 86 :57
37A0 07 AD CB 38 30 CB 62 E6 8A 10 3D EB 59 A6 C4 :CF
37B0 1F 02 34 10 A6 A4 30 CB C6 6E 59 AB 85 A7 85 A6 :08
37C0 A4 30 CB D0 E6 5A AB 85 A7 85 A6 A4 35 10 AD CB :CB
37D0 3B ED 81 C6 20 E7 80 6C 5A 59 E6 59 C1 10 26 CB :85
37E0 20 3B ED 81 31 CB D0 E6 5A 6A A5 AD CB 38 ED 81 :38
37F0 AD CB 23 6F 59 86 06 AD CB 38 30 CB 62 CC 20 20 :FF

```

Sum: 00 5B A4 07 3F DB 37 BE BB 51 61 DB 0B C9 0B 6F :45

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3800 ED 01 E6 5A 86 10 3D EB 59 A6 C4 1F 02 A6 A4 :17
3810 03 71 A7 80 6C 59 E6 59 C1 10 26 E6 AD CB 23 AD :C1
3820 CB 26 6C 5A E6 5A C1 10 26 FF 4F 86 07 AD CB :48
3830 38 30 CB 62 CC 20 05 A7 80 5A 2B FR CC 2D 31 A7 :96
3840 80 5A 2B FR AD CB 23 AD CB 26 86 05 AD CB 38 30 :F6
3850 CB 62 CC 20 73 ED 81 CC 75 6D ED 81 CC 3A 20 E7 :A6
3860 81 6F 59 E6 59 34 10 3D CB C0 A6 85 AB CB ED A7 :A9
3870 CB 0E A6 85 35 10 AD CB 38 ED 81 86 20 A7 80 6C :57
3880 59 E6 59 C1 10 26 CC 20 38 ED 81 A6 CB ED AD :FB
3890 CB 38 ED 81 AD CB 23 AD CB 26 AD CB 26 59 E6 41 :9F
38A0 86 10 34 04 3D 35 04 C4 0F E7 5B A7 5C 6F 5D 6F :97
38B0 5E 86 13 A0 5C AD CB 44 86 03 E6 5B 8D 1F 9B 8B :F5
38C0 06 AD CB 3E 86 07 AD CB 38 3D CB 8E 2C 81 0D 26 :53

```

```

38D0 B6 12 A0 5C AD CB 47 AD CB 26 39 B1 1B 26 10 86 :B9
38E0 02 E6 5B 3D 1F 98 40 8B 35 6D 5D 26 01 4C 63 :5E
38F0 27 05 AD CB 3E 20 D2 AD CB 41 20 CD B1 1C 26 35 :6C
Sum: 3B B4 AF A1 3B 33 1B 9A 64 42 02 CB CB 3D E4 6F :2A

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3900 6D 5D 26 03 A7 5C 8C E7 5D A6 A4 17 01 42 A6 50 :47
3910 E6 5B C1 10 27 04 86 02 20 16 6F 5B 86 2E 4D :E6
3920 6C 5B E6 5B C1 10 26 06 6F 5B 86 0F 20 3F 86 01 :4A
3930 AD CB 3E 20 94 81 1D 26 3A 6D 5E 26 1E 6D 5D 63 :A1
3940 5D 26 04 86 01 20 26 6A 5B E6 5B C1 FF 27 04 86 :CB
3950 02 20 1A C6 0F E7 5B 86 2E 20 D5 6A 5B E6 5B C1 :C3
3960 FF 26 0B C6 0F E7 5B 86 0F 20 C5 86 01 AD CB 41 :FB
3970 1A FF 56 81 1E 26 1B 6A 5C E6 5C C1 FF 26 0A 86 :04
3980 0F E7 5C 86 0F 20 C6 01 AD CB 44 16 FF 3A B1 :33
3990 1F 26 16 6C 5C E6 5C C1 10 26 06 6F 5C 86 20 :E2
39A0 FB 86 01 AD CB 47 16 FF 20 34 02 86 10 E6 5C 3D :AB
39B0 EB 5B A6 C4 1F 02 35 02 6D 5E 10 26 00 ED 81 30 :A7
39C0 25 0E B1 39 23 0D 80 07 B1 3A 25 04 B1 3F 23 03 :A6
39D0 1A FE 84 0F E6 A4 6D 5D 26 07 4B 4B 4B 4A 34 :02
39E0 0F 9C C4 F0 34 04 A0 E6 A4 A7 A4 A6 A4 34 04 :68
39F0 1F 89 E0 E0 E7 5F 17 01 8A A7 5B A6 A4 AD CB 3B :49

```

Sum: 4A 56 BE 2A C5 AB CB 0B D0 FE 35 FB D3 0C 2B 6E :18

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3A00 6D 5D 26 03 A7 5C 8C E7 5D A6 A4 17 01 42 A6 50 :47
3A10 AD CB 29 86 01 6D 5D 26 01 81 4C AD CB 41 30 5C :25
3A20 86 11 A0 84 AD CB 47 AD CB 46 52 AD CB 29 A6 53 AD :7B
3A30 29 86 03 E6 5B 3D 1F 98 40 8B 30 AD CB 3E A6 56 :91
3A40 AD CB 29 A6 57 AD CB 29 86 02 AD CB 41 A6 5C 40 :B9
3A50 8B 11 AD CB 44 A6 54 AD CB 29 A6 55 AD CB 29 A6 :2C
3A60 5B 8B 02 AD CB 3E A6 58 AD CB 29 A6 58 AD CB 40 8B :48
3A70 35 6D 5D 26 01 4C AD CB 41 63 5D 10 26 FE A6 C6 :D2
3A80 5B E6 5B C1 10 27 08 B6 01 AD CB 3E 1E FE 3A 6F :93
3A90 5B 6C 5C 0C 10 26 05 6A 5C 1E FE 32 86 2F :22
3AA0 AD CB 41 86 01 AD CB 47 16 FE 1F E6 4A 47 A6 A6 :A6
3AB0 A4 34 04 1F 89 E0 E7 5F AD CB 3B ED 50 A6 A4 :C1
3AC0 17 00 C0 A7 5B 17 00 8B A6 5B AD CB 29 A6 5B 8B :9D
3AD0 05 AD CB 41 A6 54 AD CB 29 A6 55 AD CB 29 86 02 :74
3AE0 AD CB 41 86 11 A0 5C AD CB 47 A6 56 AD CB 29 A6 :45
3AF0 57 AD CB 29 A6 5B C6 03 3D 1F 98 40 8B 34 AD CB :27

```

Sum: 8B FD B4 17 BF 7A 7D 1B 6E DB 0B ED 74 BA F9 BA :13

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3B00 41 A6 52 AD CB 29 A6 53 AD CB 29 86 02 AD CB 41 :AC
3B10 86 11 A0 5C AD CB 44 A6 5D AD CB 29 A6 51 AD CB 4C :48
3B20 29 A6 5B 48 40 8B 35 AD CB 3E 6C 5B E6 5B C1 10 :FE
3B30 10 26 FD 95 6F 5B 6C 5C E6 5C C1 10 26 05 6A 5C :5E
3B40 16 FD 8D 86 10 AD CB 41 86 01 AD CB 47 16 FD 79 :BB
3B50 30 CB D0 A6 5C 30 86 A6 5F AB 84 A7 84 AD CB 38 :8F
3B60 ED 54 30 CB 0E A6 5F AB 84 A7 84 AD CB 38 ED 56 :6B
3B70 30 CB 0C A6 5B 30 86 A6 5F AB 84 A7 84 AD CB 38 :7E
3B80 ED 52 39 81 7F 27 09 B1 20 25 05 81 FB 24 01 8C :9D
3B90 86 2E 39 00 00 00 00 00 00 00 00 00 00 00 00 :ED
3BA0 16 04 A7 80 3A 41 42 58 20 91 89 95 99 9A 89 9D :AE
3BB0 A9 41 44 43 41 20 91 C9 95 D9 9A F9 9D E9 41 44 :38
3BC0 43 42 20 91 8B 95 9B 9A 8B 9D AB 41 44 44 41 :88
3BD0 91 CB 95 DB 9A FB 9D EB 41 44 44 42 02 92 C3 95 :FE
3BE0 D3 9A F3 9D E3 41 44 44 44 20 91 84 95 9A 9A :89
3BF0 9D B4 41 4E 44 41 20 91 C4 95 D4 9A F4 9D E4 41 :93

```

Sum: D9 B4 DD 1B 11 24 36 36 4C 32 D3 BD E6 B7 97 C1 :C9

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3C00 4E 44 42 20 91 C1 4E 44 43 43 40 8B 48 41 53 4C :A2
3C10 41 20 80 5B 41 53 4C 42 20 95 08 9A 7B 9D 68 41 :70
3C20 53 40 20 80 47 41 53 52 41 20 80 57 41 53 42 :CC
3C30 20 95 07 9A 77 9D 67 41 53 52 41 91 85 95 95 :B1
3C40 A5 9D 85 42 49 54 41 20 91 C5 95 D5 9A 95 9D :08
3C50 42 49 54 42 20 80 4F 43 4C 52 41 20 80 5F 43 :C0
3C60 52 42 20 95 0F 9A 7F 9D 6F 43 4C 52 20 91 81 :95
3C70 91 9A 81 9D 41 43 40 50 41 20 91 C1 95 D1 9A 91 :9E
3C80 9D E1 43 40 50 42 20 82 B3 85 93 BA 83 8D A3 43 :4D
3C90 4D 50 44 20 92 8C 95 9C 9A 8C 9D AC 43 4D 50 :58
3CA0 20 82 8C 85 9C BA 8C 9D AC 43 4D 50 59 20 D2 8C :45
3CB0 D5 9C BA 8C 9D AC 43 4D 50 53 20 D2 83 D5 93 :7A
3CC0 83 D0 A3 43 4D 50 55 20 80 43 43 4F 4D 41 20 80 :08
3CD0 53 43 4F 4D 42 20 95 03 9A 73 9D 63 43 4F 4D :38
3CE0 91 3C 43 57 41 49 20 90 19 44 41 41 20 80 4A :9E
3CF0 45 43 41 20 80 5A 44 45 43 42 20 95 0A 9A 7A 9D :41

```

Sum: 87 25 26 2D 54 45 A5 B3 14 07 7C 1A E1 25 26 A2 :6F

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3D00 6A 44 45 43 20 91 88 95 9B 9A 8B 9D AB 45 4F :52
3D10 41 20 91 CB 95 DB 9A FB 9D EB 45 4F 52 42 20 85 :0B
3D20 1E 45 58 47 20 80 4C 49 4E 43 41 20 80 5C 49 :9C
3D30 43 42 20 95 0C 9A 7C 9D 6C 49 4E 43 20 95 0E :9A
3D40 7E 9D 6E 4A 4D 50 20 95 9D 9A 8D 4D A4 53 52 :52
3D50 20 91 86 95 9A 9A 9D 6A 4C 44 41 20 91 C6 95 :02
3D60 D6 9A F6 9D E6 4C 44 42 20 92 CC 95 DC 9A FC 9D :DD
3D70 EC 4C 44 44 20 92 8E 95 9E 9A BE 9D AE 44 5A :8E
3D80 20 82 8E 85 9E BA 8E 8D AE 44 44 44 44 59 20 :82
3D90 DE BA FE 8D EC 44 43 20 92 CE 95 DE 9A FE 9D :4C
3DA0 EE 4C 44 55 20 9D 3C 4C 45 41 5B 20 9D 31 4C :69
3DB0 41 59 20 9D 32 4C 45 41 53 20 9D 33 4C 45 41 :55
3DC0 20 80 48 4C 53 4C 41 20 80 5B 4C 53 4C 42 20 :95
3DD0 0B 9A 7B 9D 6B 4C 53 4C 20 80 4A 44 4C 53 52 :40
3DE0 80 54 4C 53 52 42 20 95 0A 7A 9D 6A 4C 53 52 :20
3DF0 20 80 3D 4D 55 4C 20 80 40 4E 45 47 41 20 80 :86

```



## SUPER MONITOR ダンプ・リスト

Sum:	61	FE	B5	94	0A	60	DD	9A	3A	1F	67	23	1C	FB	AC	DE	:0D
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3E00	4E	45	47	42	20	95	00	9A	70	9D	60	4E	45	47	20	80	:52
3E10	12	4E	4F	50	20	91	8A	95	9A	9A	8A	9D	0A	4F	52	41	:E6
3E20	20	91	CA	95	DA	9A	FA	9D	EA	4F	52	42	20	91	1A	4F	:02
3E30	52	43	43	20	85	34	50	53	48	53	20	85	36	50	53	48	:B5
3E40	55	20	85	35	50	55	4C	53	20	85	37	50	55	4C	55	20	:B5
3E50	80	49	52	4F	4C	41	20	80	59	52	4F	4C	42	20	95	09	:D0
3E60	9A	79	9D	69	52	4F	4C	20	80	46	52	4F	52	41	20	80	:C0
3E70	56	52	4F	52	42	20	95	06	9A	76	9D	66	52	4F	52	20	:6C
3E80	B0	38	52	54	49	20	80	39	52	54	53	20	91	82	95	92	:06
3E90	9A	B2	9D	A2	53	42	43	41	20	91	C2	95	D2	9A	F2	9D	:A7
3EAA	E2	53	42	43	42	20	80	1D	53	45	58	20	95	97	9A	B7	:46
3EAB	9D	A7	53	54	41	20	95	07	9A	F7	9D	E7	53	54	42	20	:D6
3EAC	95	DD	9A	FD	9D	ED	53	54	44	20	95	9F	9A	BF	9D	AF	:77
3EDA	53	54	58	20	85	9F	BA	BF	BD	AF	53	54	59	20	85	DF	:0C
3EE0	BA	FF	BD	EF	53	54	53	20	95	DF	9A	FF	9D	EF	53	54	:BF
3EF0	55	20	91	80	95	90	9A	B0	9D	A0	53	55	42	41	20	91	:0E
Sum:	27	D2	2A	9F	28	0B	F3	69	61	DB	E0	06	9D	89	63	9A	:96
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3F00	C0	95	D0	9A	F0	9D	02	95	53	55	42	4C	20	92	83	95	:85
3F10	9A	B3	9D	A3	53	55	42	44	20	80	37	53	57	49	20	A0	:4D
3F20	3F	53	57	49	32	20	C0	3F	53	49	33	20	80	13	53	AF	:A0
3F30	59	4E	43	20	85	1F	54	46	52	20	80	4D	54	53	54	41	:C3
3F40	20	80	5D	54	53	54	42	20	95	0D	9A	7D	9D	6D	54	53	:C4
3F50	54	20	89	24	42	43	43	20	89	24	42	48	53	20	89	25	:61
3F60	42	43	53	20	89	25	42	4C	4F	20	89	27	42	45	51	20	:48
3F70	49	2C	42	47	45	20	89	2E	42	47	54	20	89	22	42	48	:8C
3F80	49	20	89	2F	42	4C	45	20	89	23	42	4C	53	20	89	2D	:77
3F90	42	4C	54	20	89	2B	42	4D	49	20	89	26	42	4E	45	20	:52
3FA0	49	2A	42	50	4C	20	89	2D	42	52	41	20	89	21	42	52	:8D
3FB0	4E	20	89	8D	42	53	52	20	89	28	42	56	43	20	89	29	:E9
3FC0	42	56	53	20	AE	24	4C	42	43	43	20	AE	24	4C	42	48	:B9
3FD0	53	20	AE	25	4C	42	43	53	20	AE	25	4C	42	4C	4F	20	:A6
3FE0	AE	27	4C	42	45	51	20	AE	2C	42	47	45	20	AE	2E	09	:09
3FF0	4C	42	47	54	20	AE	22	4C	42	48	49	20	AE	2F	4C	42	:C3
Sum:	22	BD	BE	BC	15	5C	B9	12	37	13	21	48	D2	29	80	47	:DA
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4000	4C	45	20	AE	23	4C	42	4C	4C	53	20	AE	20	4C	42	4C	:D8
4010	20	AE	28	4C	42	4D	40	AE	26	4C	42	4E	45	20	AE	00	:04
4020	2A	4C	42	50	4C	20	8E	16	4C	42	52	41	20	AE	21	4C	:74
4030	42	52	4E	20	BE	17	4C	42	53	52	20	AE	28	4C	42	56	:B4
4040	43	20	AE	29	4C	42	56	53	20	FF	EC	C4	26	4C	EC	46	:9C
4050	ED	C4	86	05	AD	C8	38	AD	C8	26	30	C8	62	A6	C4	AD	:F5
4060	CB	38	ED	01	A6	41	AD	C8	38	ED	01	CC	20	20	ED	81	:F0
4070	ED	81	AD	C8	23	86	07	AD	C8	38	30	C8	62	CC	20	48	:CE
4080	A7	80	5A	26	FB	6F	5B	6F	5A	6F	5C	A6	D4	AD	C8	38	:2A
4090	30	CB	62	ED	81	10	83	31	30	27	08	10	83	31	31	27	:07
40A0	05	20	15	C6	01	8C	C6	01	ED	5B	EC	C4	C3	00	01	ED	:F8
40B0	C4	AE	D4	AD	C8	38	ED	01	AF	50	11	8D	FA	E5	A6	A0	:3E
40C0	2A	FC	B1	FF	27	53	AE	A0	A1	D4	26	F2	A6	3E	C6	08	:A5
40D0	3D	84	03	A1	5B	26	E7	A6	3E	E6	3E	44	44	84	07	A7	:8F
40E0	5D	C4	03	E7	5C	30	C8	72	AE	A1	2B	FC	31	3F	A7	80	:D6
40F0	A6	A0	28	04	A7	80	20	FB	AF	52	A6	5D	10	27	02	DD	:CE
Sum:	C7	23	00	F2	CB	10	AD	0C	DF	12	EF	14	2B	02	A2	5B	:BE
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4100	4A	27	72	4A	10	27	01	2C	4A	10	27	01	2E	4A	27	3A	:EC
4110	4A	27	3C	4A	27	3E	16	01	32	86	0A	AD	C8	38	30	CB	:D4
4120	72	CC	46	43	ED	81	CC	42	20	ED	81	5D	04	86	24	A7	:56
4130	B0	31	C8	62	EC	A1	81	20	27	09	ED	81	CC	2C	24	ED	:80
4140	81	20	F1	30	1E	AF	52	16	02	93	CC	23	24	20	08	CC	:93
4150	3C	24	20	03	CE	02	24	30	AD	C8	7A	10	AE	C4	31	21	:ED
4160	B1	E6	50	E7	5E	A6	A0	AD	C8	38	ED	81	6A	5E	26	F5	:4F
4170	AF	52	16	02	93	00	AE	C4	A6	84	81	1E	27	64	81	1F	:27
4180	60	E6	01	E7	5E	30	AE	C4	A6	84	81	1E	27	64	81	1F	:27
4190	06	A7	A4	A6	C8	73	81	53	07	08	86	07	A6	5E	BD	11	:25
41AA	4A	2A	F9	20	5B	4F	68	5E	8D	07	4C	81	08	26	F7	20	:A5
41B0	4F	25	01	39	31	BD	00	19	31	A6	31	A6	E6	A0	C1	45	:BF
41CA	7A	04	32	62	20	41	E7	80	06	A0	E7	80	C6	2C	E7	80	:CC
41D0	39	50	43	53	02	59	02	58	02	44	50	42	02	41	02	43	:D4
41E0	43	E6	01	1F	98	54	54	54	54	58	84	0F	48	30	C8	7A	:D6
41F0	31	8D	00	20	31	A5	BD	C4	31	BD	00	18	31	A6	BD	BC	:FB
Sum:	45	6A	54	2F	45	9E	B9	5C	5C	63	3E	36	B5	93	05	62	:0C
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4200	30	1F	AF	52	16	01	BE	AE	C4	A6	01	AD	C8	38	ED	CB	:A3
4210	64	16	FF	05	04	02	58	02	59	02	55	02	53	02	50	43	:8B
4220	45	02	45	02	41	02	42	02	43	43	44	50	45	02	45	02	:8D
4230	45	02	45	02	42	AE	C4	E6	01	10	20	04	AE	C4	EC	01	:E4
4240	54	30	CB	7A	AF	52	63	5A	16	01	7A	10	AE	C4	E6	21	:9E
4250	86	08	30	84	03	27	0F	4A	27	09	4A	27	03	86	53	8C	:D8
4260	86	55	8C	86	59	8C	86	5B	A7	59	E6	21	10	2A	01	28	:1A
4270	C4	1F	31	8D	00	0C	5D	10	27	00	9B	6D	2A	FC	5A	:66	
4280	20	F4	3C	5B	28	0B	2C	5B	28	0B	2C	5B	28	0B	2C	5B	:08
4290	2D	2D	5B	28	0B	2C	5B	28	0B	2C	5B	28	0B	2C	5B	28	:08
42AA	DB	5A	2C	5B	28	0B	2C	5B	28	0B	2C	5B	28	0B	2C	5B	:5F
42B0	2C	50	43	52	20	DB	57	2C	5D	43	52	20	DB	57	2C	5D	:04
42C0	DE	45	DB	58	2C	5B	28	0B	2C	5B	28	0B	2C	5B	28	0B	:49
42D0	58	5D	DB	58	2C	5B	28	0B	2C	5B	28	0B	2C	5B	28	0B	:9C
42E0	2C	58	5D	DB	58	2C	5B	28	0B	2C	5B	28	0B	2C	5B	28	:7F
42F0	5D	DB	58	2C	5B	28	0B	2C	5B	28	0B	2C	5B	28	0B	2C	:52
Sum:	52	81	45	B5	9E	0E	2C	91	48	C9	EC	A7	24	C4	1C	4D	:12E
Add	+0	+1	+2	+3													

4320	5D	DR	DB	30	CB	7A	A6	A0	10	2B	00	A5	B1	56	27	22	:CB
4320	B1	57	27	75	B1	50	27	4F	B1	58	27	45	B1	57	04	16	:A1
4330	B7	57	20	E2	AE	C4	A6	27	01	AD	C8	3B	10	ED	B1	16	:D1
4340	FD	D7	34	22	CC	02	24	A7	5C	E7	C8	3B	10	AE	C4	A6	:22
4350	FD	D7	34	22	CC	02	24	A7	5C	E7	C8	3B	10	AE	C4	A6	:22
4360	E7	80	18	3E	E4	A6	22	AD	C8	3B	ED	B1	A6	23	AD	C8	:00
4370	3B	ED	B1	56	25	04	A6	22	AD	C8	3B	ED	B1	A6	23	AD	:C8
4380	B7	58	00	35	25	04	22	1D	CB	0F	7A	4F	ED	54	25	20	:77
4390	B5	A6	59	A7	80	18	F2	7E	50	40	3A	1F	B1	0A	25	02	:B6
43A0	D7	C6	D0	E7	80	18	F2	7E	50	40	3A	1F	B1	0A	25	02	:B6
43B0	AD	C8	3B	ED	B1	30	27	02	A7	80	E7	80	96	2C	E5	59	:ED
43C0	B1	AF	52	20	00	10	A6	C4	31	21	E6	5C	ED	5A	50	:F6	
43D0	20	02	AE	AD	AD	C8	3B	ED	B1	C4	5D	26	F5	AE	52	:AD	
43E0	C8	33	4F	E6	5C	5C	E5	C4	ED	C4	ED	46	B1	5A	27	:72	
43F0	B6	05	AD	C8	3B	30	C8	62	CC	4C	3A	ED	B1	EC	46	:E3	
Sum:	CC	47	0B	34	F4	CA	6F	A7	1C	B6	71	75	15	B2	53	D9	:CE
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4400	54	74	04	AD	C8	3B	ED	B1	35	02	AD	C8	3B	ED	B1	AD	:AD
4410	C8	23	6A	45	10	26	FC	3A	ED	AD	C8	26	39	00	00	00	:DA
4420	EC	C4	ED	46	AD	C8	26	B6	05	AD	C8	3B	ED	B1	C8	62	:AE
4430	C4	ED	C8	3B	ED	C8	26	B6	05	AD	C8	3B	ED	B1	C8	62	:AE
4440	B6	07	AD	AD	C8	3B	ED	4F	33	C2	62	0F	12	AD	80	5A	:4D
4450	26	FF	AF	AD	AD	C8	32	AD	A4	19	26	62	0F	AD	C8	26	:21
4460	C6	19	30	CB	C0	86	20	A7	80	5A	26	FB	30	CB	71	31	:79
4470	C9	00	B2	34	20	A6	80	80	30	26	A6	80	30	26	FB	C6	:ED
4480	05	31	CB	C0	8D	30	26	80	30	26	A6	80	30	26	FB	C6	:ED
4490	25	26	1F	AD	21	26	04	A6	80	A7	A0	A6	80	5A	26	F5	:1C
44A0	BD	47	AD	A0	26	04	A6	80	5A	26	F7	8D	0B	20	04	32	:62
44B0	17	32	62	16	03	36	20	AD	C7	0F	AD	C6	22	07	B1	20	:39
44C0	32	62	20	E2	33	62	20	E9	6F	53	30	CB	62	86	1E	A7	:A1
44D0	80	CC	C	06	A7	80	5A	26	FB	C0	20	0B	A7	80	5A	26	:AE
44E0	FB	6F	04	33	CB	C0	20	0B	AD	C8	C0	C1	20	26	04	AD	:C6
44F0	39	10	83	46	43	26	11	EC	EC	C0	C2	10	B3	43	20	27	:E0
Sum:	BB	2D	53	23	72	3E	F0	F3	3C	32	13	55	BE	0F	00	2B	:B9
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4500	10	B3	42	20	27	25	20	20	10	B3	42	53	26	0B	EC	CB	:BE
4510	C8	12	03	5A	20	27	25	20	0F	10	B3	52	44	26	09	EC	:BF
4520	C8	12	03	5A	20	27	25	16	00	B3	30	CB	C5	17	03	:91	
4530	B1	17	22	02	8D	6D	50	27	07	6D	59	26	03	16	02	B5	:FC
4540	A6	51	17	02	C7	A6	1F	B1	20	26	E3	16	02	95	30	CB	:EB
4550	C5	06	80	34	02	A6	B0	A1	E4	27	0B	17	02	AE	5A	26	:42
4560	F4	23	05	32	61	16	02	7B	32	61	16	02	7F	30	CB	C5	:26
4570	17	03	1F	10	25	02	7B	47	17	02	91	AE	50	30	1F	10	:41
4580	A0	10	02	0A	20	50	6D	10	26	02	6A	20	EA	30	CB	C5	:AE
4590	17	02	FF	10	25	02	3B	A6	51	B1	0A	26	04	6F	4E	20	:33
45A0	0A	B1	10	00	16	02	4B	F4	49	63	4E	16	02	31	30	8D	:96
45B0	F5	31	31	CB	C0	86	2A	09	43	1E	27	02	31	30	02	:DB	
45C0	20	F3	AF	55	AE	80	2B	0C	A1	AD	27	F8	6D	B4	2B	ED	:02
45D0	30	01	20	FE	AE	55	A6	1E	49	49	49	84	03	27	05	:E7	
45E0	8R	0F	17	02	27	A6	1E	44	44	84	07	27	3D	4A	27	:AD	
45F0	4A	10	27	00	BC	4A	10	27	00	B6	30	CB	C5	A6	B4	:A1	
Sum:	D3	0B	62	5D	0D	19	B2	9C	B5	FC	AC	BE	ED	2B	F2	56	:F6
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4600	20	10	27	01	E7	C6	0A	B6	80	B1	2C	10	27	01	10	5A	:B5
4610	26	F5	A6	CB	C5	B1	20	19	20	00	B6	B1	3C	11	27	00	:07
4620	B7	B1	5B	10	27	00	F9	16	00	B5	A6	CB	C5	B1	20	10	:72
4630	26	02	30	A1	AF	55	17	01	D4	16	01	A8	A6	1F	17	01	:CC
4640	6F	50	A6	1F	30	CB	C5	B1	E7	27	04	1F	26	17	17	:FF	
4650	01	DC	10	25	01	9C	B6	10	3D	E7	50	17	01	10	10	:25	
4660	01	90	AA	50	20	15	A6	B4	B1	20	27	0D	17	01	10	:16	
4670	25	01	7F	AE	57	50	20	ED	A6	50	17	01	BF	16	01	:A6	
4680	62	A6	1F	17	01	B6	6C	53	30	CB	C5	17	02	04	10	:01	
4690	01	60	EC	57	A3	C4	B3	00	01	10	B3	FF	B0	24	09	:00	
46A0	B3	00	7F	23	03	16	01	4D	1F	9B	17	01	5F	16	01	:33	
46B0	A6	1F	17	01	57	6C	53	30	CB	C5	17	01	D5	10	25	:01	
46C0	31	EC	50	A4	C4	B3	00	02	34	04	17	01	5F	16	02	:17	
46D0	01	3A	16	01	0E	B6	10	8C	B6	14	30	CB	C4	20	05	:B6	
46E0	1B	30	CB	C5	A7	57	17	01	A9	10	25	01	05	AE	55	:A6	
46F0	B3	10	2A	00	FD	4A	1C	A1	57	26	F4	A6	01	17	01	:0C	
Sum:	12	CF	B9	F1	07	6E	2E	D5	5B	8E	D5	43	40	96	F1	3B	:03
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4700	6C	53	A6	B4	85	05	26	0B	AD	50	27	0C	6D	59	26	:B8	:B5
4710	16	00	E2	A6	1F	17	00	F4	A6	51	17	00	EF	16	00	C3	:FC
4720	AE	55	AE	1F	17	00	55	6F	A6	51	CB	C5	A6	B4	81	55	:8F
4730	26	02	30	A1	AF	55	17	00	F5	24	3A	AE	55	A6	B4	81	:6B
4740	2C	27	2B	17	01	AD	25	23	30	1F	3A	10	30	CB	D7	34	:80
4750	10	AE	55	30	01	34	10	AD	CB	20	6C	45	A6	59	27	04	:BF
4760	B6	56	20	04	B6	57	4C	45	A7	0B	F5	B6	80	A7	57	30	:36
4770	CB	C5	A6	20	0A	20	27	1E	B1	59	59	27	0A	B1	55	27	:0A
4780	B1	53	27	0A	20	EC	B6	A0	20	06	B6	20	02	B6	60	:2B	
4790	A7	57	B6	58	A7	1F	F1	B0	FA	ED	4F	30	CB	C5	E6	A0	:D4
47A0	2B	0F	E1	80	27	F8	E6	A0	2A	FC	AD	B1	1F	23	EC	20	:B1
47B0	A1	AA	57	1F	B9	C4	1F	C1	0B	26	14	E6	51	C1	F0	2D	:E5
47C0	0E	C1	0F	2E	0A	C4	1F	FA	57	C4	7F	1F	9B	6F	45	8D	:75
47D0	3B	6D	45	27	0E	6C	53	6A	45	27	04	A6	50	8D	20	A6	:11
47E0	51	8D	29	EC	C4	ED	46	31	C8	62	20	14	B6	53	8C	B6	:A4
47F0	43	8C	B6	4F	BC	B6	56	31	8C	0C	A7	A4	EC	46	ED	C4	:03
Sum:	51	44	89	A6	B3	CF	FA	E5	A9	CE	71	3B	A7	F6	DA	63	:A9
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4800	AD	C8	35	16	FC	1E	20	2D	45	52	52	00	16	E6	53	:93	
4810	C1	0A	04	24	D0	30	C8	69	3A	AD	C8	3B	ED	B4	63	53	:E3
4820	53	A6	E4	A7	AD	C4	AE	C4	30	01	AF	C4	35	16	39	34	:46
4830	31	8C	36	EC	B1	C1	2C	27	06	C1	A0	27	02	20	03	C6	:60
4840	20	8C	30	01	10	A3	AA	27	08	6D	AA	27	17	31	24	20	:27

```

4850 F3 E6 1F C1 20 27 06 C1 2C 27 02 20 07 EC 22 1C :6D
4860 FE 35 20 39 1A 01 35 20 39 50 43 05 80 53 20 04 :C4
4870 40 55 20 03 40 59 20 02 20 58 20 01 10 44 20 00 :B0
4880 00 44 50 08 08 42 20 09 04 41 20 08 02 43 43 0A :11
4890 01 00 6F 58 A6 80 81 24 27 12 81 26 27 77 81 2D :BF
48A0 F6 04 63 58 20 EE 30 1F 6D 4E 27 69 6F E2 6F E2 :2F
48B0 A6 80 8D 23 25 33 81 30 25 56 81 39 23 0A 81 41 :03
48C0 25 4E 81 46 22 4A 8B 09 C6 10 3D 86 04 58 69 61 :F9
48D0 69 E4 4A 26 F8 20 D9 81 20 27 08 81 2C 27 07 81 :DD
48E0 5D 27 03 1C FE 39 1A 01 39 CC 00 00 6D 58 27 03 :E9
48F0 A3 E1 8C E3 E1 ED 50 6F 59 10 83 FF 80 2D 0E 10 :36

```

Sum: 9E 02 0B FD F7 EC 98 3E BB D0 8E 6C 56 39 4F 34 :F8

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4900 83 00 80 2C 08 A6 1F 81 5D 27 02 63 59 1C FE 39 :12
4910 32 62 1A 01 39 6F E2 6F E2 A6 80 8D 8A 25 CA 81 :67
4920 30 25 ED 81 39 22 E9 EC E4 58 49 58 49 E3 :9D
4930 E4 E3 E4 ED E4 E6 1F C4 0F 4F E3 E4 ED E4 20 D9 :34
4940 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4950 81 42 26 06 4F 48 63 48 20 06 81 54 26 18 6F 48 :4A
4960 17 02 4C EC C4 ED 48 17 02 2E 31 8C 04 AD C8 50 :17
4970 39 4F 2E 4B 2E 00 AD C8 53 1A 40 EF 8C 21 1F 41 :4D
4980 32 8C 7E FC FF F6 34 16 34 41 30 8D 00 CF BF FF :36
4990 F6 6F 4F 6F C8 10 63 C8 10 1C BF AD D4 20 62 00 :14
49A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
49B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
49C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
49D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
49E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
49F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00

```

Sum: C2 F8 DB 43 86 5B F8 AB EB 1F 8F 35 D3 52 AB 51 :42

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4A00 00 32 8C F6 35 41 35 16 FD FF F6 1F 14 1C BF 17 :8C
4A10 01 9D 6D 4F 2B 07 26 1A 31 8C 1E 20 18 31 8C 20 :8C
4A20 A6 8C 3B AD C8 3B ED A4 A6 8C 31 AD C8 3B ED 22 :CD
4A30 20 03 31 8C 1A AD C8 50 39 5F 65 74 75 72 6E 00 :78
4A40 5D 2A 2A 2A 2D 4B 65 79 20 42 72 61 6B 65 00 42 :45

```

```

4A50 72 61 6B 65 20 50 6F 69 6E 74 00 00 00 EE 8D FF :47
4A60 3E 86 FF A7 4F 6D C8 10 27 08 6F C8 10 EC 61 ED :AE
4A70 8C E9 30 8C 8C AF 61 3B EE 8D FF 23 AD C8 26 17 :57
4A80 01 2D EC 48 ED 6A ED C4 30 C8 62 CC 20 20 ED 81 :3E
4A90 CC 50 43 ED 81 86 3A A7 80 86 0A 8D 2B CC 53 3A :55
4AA0 ED 81 31 6C 34 20 4F 8D 1F 32 62 CC 55 3A ED 81 :B7
4AB0 86 08 8D 14 CC 59 3A ED 81 86 06 8D 0B CC 58 3A :7E
4AC0 ED 81 86 04 8D 02 20 23 4C 4C EC E6 34 04 AD C8 :E1
4AD0 3B ED 81 35 02 AD C8 3B ED 81 CC 20 20 ED 81 39 :B1
4AE0 4C 4C A6 E6 AD C8 3B ED 81 20 EF CC 44 50 ED 81 :1F
4AF0 B6 3A A7 80 86 03 8D EB CC 41 42 ED 81 86 3A A7 :09

```

Sum: 67 52 67 94 9A CA 6D 69 86 E8 47 1D 55 BA 94 3D :AD

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4B00 80 86 01 8D C3 CC 43 43 ED 81 86 3A A7 80 86 00 :84
4B10 8D CE AD C8 23 86 01 A7 45 17 FO 84 86 07 AD C8 :F3
4B20 38 6D 4B 27 07 86 01 A7 4F 16 FE D5 AE 48 A6 84 :A4
4B30 81 7E 26 04 EC 01 20 5B 81 16 26 06 EC 46 E3 01 :6A
4B40 20 51 81 8D 27 48 81 17 27 4F A6 5D 81 02 27 06 :AA
4B50 81 03 27 20 3B A6 84 A7 8C 0E A6 E4 1F A9 84 :67
4B60 0F C4 F0 34 04 AA E0 1F 8A 00 02 20 24 E6 01 1D :78
4B70 E3 46 20 1F A6 01 A7 8C 0E A6 E4 1F A9 84 0F C4 :F9
4B80 F0 34 04 AA E0 1F 8A 00 02 20 06 EC 02 E3 46 20 :BA
4B90 02 EC 46 ED 48 8D 01 3B A6 D8 0B 31 3F 27 07 A7 :4D
4BA0 4A 86 3F A7 D8 0B 31 8D FE CE 10 BF FF FA 39 A6 :C7
4BB0 D8 0B 81 3F 26 05 A6 4A A7 D8 0B CC 01 D7 FD FF :E2
4BC0 FA 39 00 00 00 00 00 00 00 00 00 00 00 00 00 :33
4BD0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4BE0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4BF0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00

```

Sum: 67 84 E1 FD F0 C3 75 44 B5 DB 5A D3 3A 7B 1F 24 :EA

## R A N D O M B O X

### ミシン目スキップ・プログラム

■坂本幸司

FM-8のユーザーでMP-80かMB-27401を使ってる方へ。  
デバッグ時はかまいませんがリストをファイルする場合、  
ミシン目がじゃまになります。そこで、テキスト時の66行印  
字後、1インチ紙送りしてミシン目を飛び越すようにしてや  
ります。DIPスイッチの[2-4 ON]でもできますが、いち  
いち切り替えるのもたいへんです。このプログラムはBIOSの  
LPT OUTルーチンを使って5 Byteのコントロール・コード  
を送ってます。リストの\$681Dから2 Byteが出力データ・バ  
イト数で、\$681Fから出力データです。簡単なプログラムで  
すが、以上を変更すればけっこう応用できると思います。一  
応リロケータブルなので都合がわるければ移動してください。  
頭出しをしてプリンタの電源を入れて\$6800をコールしてく  
ださい。あとは、いつものように[PFZ]を押してください。  
プリンタの電源を再投入すれば解除されます。

### ミシン目スキップ・アセンブル・リスト

```

00010      NAM      SKIP
00020      OPT      SYM
00030 6800      ORG      $6800
00040
00050      *      START      EQU      *
00060 6800 1F      51      TFR      PC, X
00070 6802 AF      8D 0015  STX      DBA, PCR
00080 6806 86      1D      LDA      #1D
00090 6808 AB      8D 0010  ADDA     DBA+1, PCR
00100 680C A7      8D 000C  STA      DBA+1, PCR
00110 6810 30      8D 0005  LEAX     RGN, PCR
00120 6814 AD      9F FBFA  JSR      [$FBFA]
00130 6818 39      RTS
00140      ***** (BIOS LPT OUT)
00150 6819      0E00      RGN     FDB     $0E00
00160 681B      0002      DBA     RMB     2
00170 681D      0005      BYT     FDB     $0005
00180 681F      1B32      DAT     FDB     $1B32
00190 6821      1B4E      FDB     $1B4E
00200 6823      06      FCB     $06
00210
00220      *      END      START

```



# MEMORY CHANGER



水戸マイコンクラブ 今泉 裕司

マシン語入力というのは実には大変なものです。その能率を少しでも上げられたらと考え、このプログラムを作りました。

## 特 長

- ①キー入力をしながら、チェック・サムでその正誤が確認できる。
- ②ASCII ダンプ機能があり、入力しているプログラムのメッセージが一目でわかる (I/O '82年9月号のp. 354, 5の意見により採用)。
- ③強力なエディット・コマンド。
- ④F-BASICのROMは一切使っていないので、どのバージョンにも適応。
- ⑤256バイト単位 (\$×00) でポジション・インディペンデント。などです。

## コマンド

### ★16進キー配列

U-MONITORと同じです。これがいやな人は\$ 306 B以後を変えてください。\$ 0～\$ Fに対応したASCII コードが入ります。

### ★ページ・アップ、ダウン

↑で一つ前のページを、↓で次のページを表示します。

### ★カーソル移動

- 1) ←でカーソルをその行の先頭に移動します。
- 2) SHIFT+ (↑, ↓, ←, →) で、カーソルを上下左右に動かすことができます。そのページの上限、下限をオーバーするとページ・アップ、ダウンを行いません。

### ★各モード変更

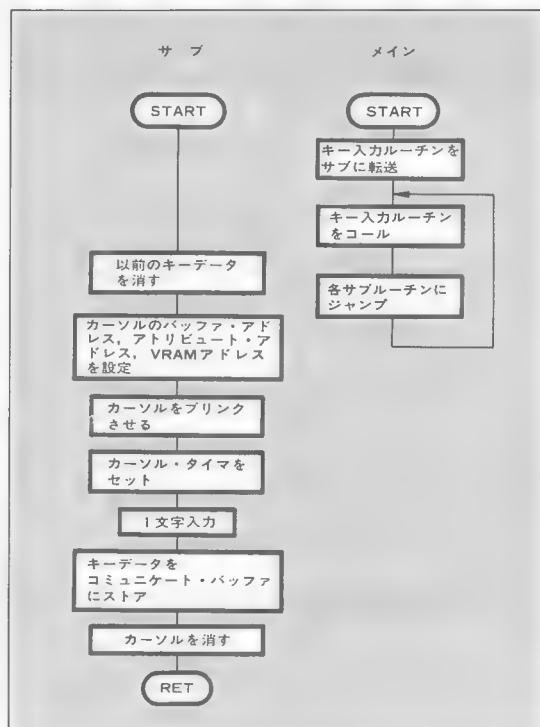
- 1) **[INS]**はF-BASICと同じ使い方をします。
- 2) **[S]**でキー入力時に音を出すかどうかを選択できます。あまり大きな音ではないのですが。
- 3) **[P]**でそのページからの抜け出しの禁止、許可を切り換えます。ただし、これはそのページの最後の1バイトの下位を入力したときに次のページに移るかどうかというスイッチです。

### ★その他のエディット・コマンド

### 16進キー配列

			F
7	8	9	A
4	5	6	B
1	2	3	C
0	E		D

### フローチャート



- 1) **[HOME]**でカーソルをページの先頭に移します。 **[CLEA R]**でそのページを\$ 00で埋めたのち、カーソルをページの先頭に移します。

- 2) 4つのキー**[EL]**, **[DUP]**, **[BS]**, **[DEL]**で、行単位の編集ができます。機能はF-BASICとほぼ同じです。

### ★アドレス入力

- 1) **[I]**でアドレスを入力することができます。16進(\$ A～\$ F)





## MEMORY CHANGER アセンブル・リスト

```

007C 31A6          : MAIN ROUTINE
007D 31A6          :
007E 31A6          :
007F 31A6 30BDFE00 MAIN :LEAX -256,PC
0080 31AA 1F10     TFR X,D
0081 31AC 1FB6     TFR A,DP
0082 31AE 8D43     BSR INIT
0083 31B0 8DCD     CMDIN :BSR INKEY%
0084 31B2 C604     LDB #4
0085 31B4 D50E     BITB -REG
0086 31B6 270F     BEQ +15
0087 31B8 C6B1     LDR #B1
0088 31BA F7FD43   STR #FD03
0089 31BD BE00B0   LDX #B0
008A 31C0 301F     LEAX -1,X
008B 31C2 26FC     BNE -4
008C 31C4 7FFD03   CLR #FD03
008D 31C7 30BDFEA0 LEAX BCGTBL,PC
008E 31C9 0F       CLRB
008F 31CC A1B0     CMFA X+
0090 31CE 2605     BNE 5
0091 31D0 1702F7   LBSR MCG0
0092 31D2 20B8     BRA CMDIN
0093 31D5 5C       INCB #16
0094 31D6 C110     CMFB -14
0095 31D8 26F2     BNE -4
0096 31DA 30BDFE9D LEAX CMDTBL,PC
0097 31DE C613     LDB #13
0098 31E0 A1B0     CMFA X+
0099 31E2 2707     BEQ CMJMF
009A 31E4 3002     LEAX 2,X
009B 31E6 5A       DECB
009C 31E7 26F7     RNE -9
009D 31F0 20C5     CMFA CMDIN
009E 31F2 1F50     CMDJMF: TFR PC,D
009F 31F4 AED4     LDX X
00A0 31F7 AD8E     JSR D,X
00A1 31F1 20BD     BFA CMDIN
00A2 31F3          :
00A3 31F3          : INITIAL
00A4 31F3          :
00A5 31F3 8504     INIT :LDA #4
00A6 31F7 770E     STA REG
00A7 31F7 30BDFE89 LEAY DINIT,PC
00A8 31FB C609     LDB #9
00A9 31FD 1700A4   LBSR HALT
00AA 3200 1700AC   LBSR MOVE
00AB 3203 1700B8   LBSR SUBS.
00AC 3206 C602     LDB #2
00AD 3208 170099   LBSR HALT
00AE 320B 1700A1   LBSR MOVE
00AF 320E 1700AD   LBSR SUBS.
00B0 3211 8D77     BSR ADRINO
00B1 3213 C67C     LDB #7C
00B2 3215 170097   LBSR MOVE
00B3 3218 1700A3   LBSR SUBS.
00B4 321B 1E12     EXG X,Y
00B5 321D 30BDFE93 LEAY DINIT,PC
00B6 3221 C609     LDB #9
00B7 3223 8D7F     BSR HALT
00B8 3225 1700B7   LBSR MOVE
00B9 3228 8650     LDA #B0
00BA 322A B7FCB4   STA #FCB4
00BB 322D 7FFCB9   CLR #FCB9
00BC 3230 1700B8   LBSR SUBS.
00BD 3233 1E12     EXG X,Y
00BE 3235 C603     LDB #3
00BF 3237 8D6B     BSR HALT
00C0 3239 8D74     BSR MOVE
00C1 323B 1700B0   LBSR SUBS.
00C2 323E C645     LDB #69
00C3 3240 8D62     BSR HALT
00C4 3242 8D6B     BSR MOVE
00C5 3244 8D78     BSR SUBS.
00C6 3246 170167   LBSR DUMFO
00C7 3249 160204   LBRA CHOME
00C8 324C          :
00C9 324C 8D56     ADRINO: BSR HALT
00CA 324E 3410     FSHS X
00CB 3250 C67C     LDB #60
00CC 3252 8D5B     BSR MOVE
00CD 3254 8D6B     BSR SUBS.
00CE 3256 8D4C     BSR HALT
00CF 3258 CFCB83   LDU #FCB83
00D0 325B 6DCA     TST X
00D1 325D 2704     BEQ ADRINI
00D2 325F 3510     ADRERR: PULS X
00D3 3261 20EB     BRA ADRINO+2
00D4 3263 3343     ADRINI: LEAU 7,U
00D5 3265 ECC1     LDD X++
00D6 3267 1083190A CMPD #B190A
00D7 326B 26F2     BNE ADRERR
00D8 326D ECC1     LDD X++
00D9 326F 8D0C     BSR BNCGO
00DA 3271 9707     STA <ADR.
00DB 3273 ECC4     LDD X
00DC 3275 8D06     BSR BNCGO
00DD 3277 84F0     ANDA #F0
00DE 3279 9708     STA <ADR.+1
00DF 327B 35C0     PULS U,PC
00E0 327D 8D0E     BNCGO :BSR BNCG1
00E1 327F 3404     FSHS B
00E2 3281 C610     LDB #10

```

```

00E3 3283 3D       MUL
00E4 3284 D707     STB <YOB1
00E5 3286 3502     PULS A
00E6 3288 8D03     BSR BNCG1
00E7 328A 9A03     ORA <YOB1
00E8 328C 39       RTS
00E9 328D 8030     BNCG1 :SUBA #B30
00EA 328F 250F     ECS BNCGER
00EB 3291 810A     CMFA #B4
00EC 3293 250A     ECS BNCGR1
00ED 3295 8007     SUBA #7
00EE 3297 810A     CMFA #B4
00EF 3299 2505     ECS BNCGEF
00F0 329B 8110     CMFA #B10
00F1 329D 2401     RCC BNCGER
00F2 329F 39       BNCGR1: RTS
00F3 32A0 3264     BNCGER: LEAS 4,S
00F4 32A2 20B8     BRA ADREFF
00F5 32A4          :
00F6 32A4          : SUB SYSTEM CONTROL
00F7 32A4          :
00F8 32A4 70FD05   HALT :TST #FD05
00F9 32A7 28FB     RM1 HALT
00FA 32A9 86B0     LDA #B80
00FB 32AB 87FD05   STA #FD05
00FC 32AE 39       RTS
00FD 32AF CFCB80   MOVE :LDU #FCB80
00FE 32B2 6FC0     CLR X+
00FF 32B4 6FC0     CLR X+
0100 32B6 A8B0     LDA X+
0101 32B8 A7C0     STA X+
0102 32BA 5A       DECB
0103 32BB 26F9     BNE -7
0104 32BD 39       RTS
0105 32BE 7FFD05   SUBS. :CLR #FD05
0106 32C1 39       RTS
0107 32C2          :
0108 32C2 8109     HEX50 :CMFA #9
0109 32C4 2702     BLS HEX51
010A 32C6 8B07     ADDA #7
010B 32C8 8B30     HEX51 :ADDA #B30
010C 32CA A7C0     STA X+
010D 32CC 39       RTS
010E 32CD 3402     HEX :FSHS A
010F 32CF C610     LDB #10
0110 32D1 3D       MUL
0111 32D2 8DEE     BSR HEX50
0112 32D4 2502     PULS A
0113 32D6 840F     ANDA #B0F
0114 32D8 20E8     BRA HEX50
0115 32DA          :
0116 32DA C602     SKIP2 :LDB #2
0117 32DC 8C       FCB #BC
0118 32DD C601     SKIP1 :LDB #1
0119 32DF 8620     STA #B20
011A 32E1 A7C0     LDA X+
011B 32E3 5A       DECB
011C 32E4 26FB     BNE -5
011D 32E6 39       RTS
011E 32E7          :
011F 32E7 DF09     ASC :STU <STROP
0120 32E9 DE0B     LDU <STR1P
0121 32EB 1FB9     TFR A,B
0122 32ED C120     CMFB #B20
0123 32EF 2402     BCC +2
0124 32F1 C620     LDB #B20
0125 32F3 C17F     BSR #B7F
0126 32F5 27FA     BEQ -6
0127 32F7 E7C0     STB X+
0128 32F9 DF0B     STU <STR1P
0129 32FB DE09     LDU <STROP
012A 32FD 20CE     BRA HEX
012B 32FF          :
012C 32FF          : DISPLAY STRING SET
012D 32FF          : INPUT: Xreg = TOP ADDRESS
012E 32FF          :
012F 32FF 33BDFD58 STRS0 :LEAU STR1,PC
0130 3303 DF0B     STU <STR1P
0131 3305 33CB05   LEAU -59,U
0132 3308 31BDFD04 LEAY SUM1,PC
0133 330C 0F0F     CLR <SUM0
0134 330E 1F10     TFR X,D
0135 3310 8DB8     BSR HEX
0136 3312 1F10     TFR X,D
0137 3314 1F9B     TFR B,A
0138 3316 8DB5     BSR HEX
0139 3318 8DC0     BSR SKIP2
013A 331A A6B4     STRS1 :LDA X
013B 331C 8DC9     BSR ASC
013C 331E A6B4     LDA X
013D 3320 9B0F     ADDA <SUM0
013E 3322 970F     STA <SUM0
013F 3324 A6B0     LDA X+
0140 3326 C602     LDB #2
0141 3328 D50E     BITB <REG
0142 332A 2604     BNE +4
0143 332C ABA4     ADDA Y
0144 332E A7A0     STA Y+
0145 3330 8DAB     BSR SKIP1
0146 3332 1F10     TFR X,D
0147 3334 C40F     ANDB #B0F
0148 3336 26E2     BNE STRS1
0149 3338 8DA3     BSR SKIP1

```

## MEMORY CHANGER アセンブル・リスト

```

014A 333A 960F LDA <SUMO
014B 333C 8DBF BSR HEX
014C 333E 209A BRA SKIP2
014D 3340 ;
014E 3340 ; LINE DISPLAY
014F 3340 ; INPUT: Xreg = TOP ADDRESS
0150 3340 ;
0151 3340 8B5D LDISP0:BSR STRS0
0152 3342 17FF5F LBSR HALT
0153 3345 33BDFC07 LEAU STR0,FC
0154 3349 10BFFC80 LDY ##FC80
0155 334D 6FA0 CLR ,Y+
0156 334F CC8064 LDD ##8064
0157 3352 EDA1 STD ,Y++
0158 3354 C64E LDB ##4E
0159 3356 E7A0 STB ,Y+
015A 3358 C003 SUBB #3
015B 335A A6C0 LDISP1:LDA ,U+
015C 335C A7A0 STA ,Y+
015D 335E 5A DECB
015E 335F 26F9 BNE LDISP1
015F 3361 960D LDA ##0D
0160 3363 A7A0 STA ,Y+
0161 3365 CC1187 LDD ##1187
0162 3368 EDA4 STD ,Y
0163 336A 16FF51 LBRA SUBS.
0164 336D ;
0165 336D ; SUM DISPLAY
0166 336D ;
0167 336D 30BDFC9F SDISP0:LEAU SUM1,FC
0168 3371 960E LDA ,REG
0169 3373 8A02 ORA #2
016A 3375 970E STA ,REG
016B 3377 8DB6 BSR STRS0
016C 3379 960E LDA ,REG
016D 337B 04FD ANDA ##FD
016E 337D 970E STA ,REG
016F 337F CC0016 LDD #22
0170 3382 8D49 BSR LOC.
0171 3384 17FF1D LBSR HALT
0172 3387 33BDFC99 LEAU STR0+4,FC
0173 338B 10BFFC80 LDY ##FC80
0174 338F 6FA0 CLR ,Y+
0175 3391 CC0064 LDD ##64
0176 3394 EDA1 STD ,Y++
0177 3396 C63B LDB ##3B
0178 3398 E7A0 STB ,Y+
0179 339A CC5375 LDD ##5375
017A 339D EDA1 STD ,Y++
017B 339F CC6D20 LDD ##6D20
017C 33A2 EDA1 STD ,Y++
017D 33A4 C637 LDB ##37
017E 33A6 A6C0 SDISP1:LDA ,U+
017F 33A8 A7A0 STA ,Y+
0180 33AA 5A DECB
0181 33AB 26F9 BNE SDISP1
0182 33AD 16FF0E LBRA SUBS.
0183 33B0 ;
0184 33B0 ; MEMORY DUMP
0185 33B0 ;
0186 33B0 CC0005 DUMP0:LDD #5
0187 33B3 8D18 BSR LOC.
0188 33B5 C610 LDB ##10
0189 33B7 D70D STB <COUNT
018A 33B9 31BDFC53 LEAU SUM1,FC
018B 33BD 6FA0 CLR ,Y+
018C 33BF 5A DECB
018D 33C0 26FB BNE -5
018E 33C2 9E07 LDY <ADR.
018F 33C4 17FF79 DUMP1:LBSR LDISP0
0190 33C7 0A0D DEC <COUNT
0191 33C9 26F9 BNE DUMP1
0192 33CB 20A0 BRA SDISP0
0193 33CD ;
0194 33CD ; CURSOR SET
0195 33CD ;
0196 33CD 1F03 LOC. ,TFF D,U
0197 33CF 10BFFC80 LDY ##FC80
0198 33D3 17FECE LBSR HALT
0199 33D6 6FA0 CLR ,Y+
019A 33D8 CC8003 LDD ##8003
019B 33DB EDA1 STD ,Y++
019C 33DD CC0512 LDD ##512
019D 33E0 EDA1 STD ,Y++
019E 33E2 FFA1 STU ,Y++
019F 33E4 CC1187 LDD ##1187
01A0 33E7 EDA4 STD ,Y
01A1 33E9 16FED2 LBRA SUBS.
01A2 33EC 10BFFC80 SETCUR:LBY ##FC80
01A3 33F0 17FEB1 LBSR HALT
01A4 33F3 6FA0 CLR ,Y+
01A5 33F5 6FA0 CLR ,Y+
01A6 33F7 8603 LDA #3
01A7 33F9 A7A0 STA ,Y+
01A8 33FB CC0312 LDD ##312
01A9 33FE EDA1 STD ,Y++
01AA 3400 960E LDA <WADR.
01AB 3402 840F ANDA ##0F
01AC 3404 C603 LDB #3
01AD 3406 3D MUL
01AE 3407 CB06 ADDB #6
01AF 3409 0D0E TST <REG
01B0 340B 2B01 BMI +1

```

```

01B1 340D 5C INCB
01B2 340E E7A0 STB ,Y+
01B3 3410 960E LDA <WADR.
01B4 3412 C610 LDB ##10
01B5 3414 3D MUL
01B6 3415 8B05 ADDA #5
01B7 3417 A7A4 STA ,Y
01B8 3419 16FEAD LBRA SUBS.
01B9 341F ;
01BA 341C 16FDD3 CADFIN:LBRA INIT+4
01BB 341F 4F BASIC:CLRA
01BC 3420 1F60 TFR A,D0
01BD 3422 30BDFC8E LEAY DINIT,FC
01BE 3424 C609 LDB #9
01BF 3428 17FE79 LBSR HALT
01C0 342B 17FEB1 LBSR MOVE
01C1 342E 0650 LDA #60
01C2 3430 B7FC34 STA #FC34
01C3 3433 CC0019 LDD #25
01C4 3436 FDFC86 STD #FC86
01C5 3439 7FFC8A CLR #FC8A
01C6 343C 17FE79 LBSR SUBS.
01C7 343F C590 FULS X,FC
01C8 3441 8601 INMPT:LDA #1
01C9 3443 9B0E MODECG:EDRA <REG
01CA 3445 970C STA <REG
01CB 3447 39 RTS
01CC 344B 8608 PAGE0:LDA #8
01CD 344E 20F7 BPA MODECG
01CE 3450 8604 SOUND:LDA #4
01CF 3453 20F3 BPA MODECG
01D0 3456 0F06 CHOME:CLR <WADR.
01D1 3459 960E LDA <REG
01D2 345B 8A80 ORA ##80
01D3 345E 970E STA <REG
01D4 3461 16FF91 LBRA SETCUR
01D5 3464 960E HEADL:LDA <WADR.
01D6 3467 84F0 ANDA ##F0
01D7 346A 970E STA <WADR.
01D8 346D 20EF BRA CHOME+2
01D9 3470 8680 CURS0:LDA #80
01DA 3473 8FDC BSR MODECG
01DB 3476 2AB2 BFL SETCUR
01DC 3479 0C06 INC <WADR.
01DD 347B 960E LDA <WADR.
01DE 347E 840F ANDA ##0F
01DF 3481 1026FF79 LBNE SETCUR
01E0 3484 960E LDA <WADR.
01E1 3487 202E BRA CURS0+6
01E2 348A 8680 CURS1:LDA #80
01E3 348D 8DCB BSR MODECG
01E4 3490 1026FF6D LBMI SETCUR
01E5 3493 0A06 DEC <WADR.
01E6 3496 960E LDA <WADR.
01E7 3499 840F ANDA ##0F
01E8 349B 810F CMPA ##0F
01E9 349E 1026FF61 LBNE SETCUR
01EA 34A1 960E LDA <WADR.
01EB 34A4 2006 BRA CURS0+6
01EC 34A7 960E CURS2:LDA <WADR.
01ED 34AA 8010 SUBA ##10
01EE 34AD 970E STA <WADR.
01EF 34B0 84F0 ANDA ##F0
01F0 34B3 81F0 CMPA ##F0
01F1 34B6 1026FF4F LBNE SETCUR
01F2 34B9 200E BRA PAGE0
01F3 34BB 960E CURSD:LDA <WADR.
01F4 34BE 8B10 ADDA ##10
01F5 34C1 970E STA <WADR.
01F6 34C4 84F0 ANDA ##F0
01F7 34C7 1026FF41 LBNE SETCUR
01F8 34CA 2008 BRA PAGE0
01F9 34CD 0A07 PAGEU:DEC <ADR.
01FA 34D0 17FEFE LBSR DUMP0
01FB 34D3 16FF37 LBRA SETCUR
01FC 34D6 0C07 PAGED:INC <ADR.
01FD 34D9 17FEF6 LBSR DUMP0
01FE 34DB 16FF3F LBRA SETCUR
01FF 34DE 9E07 CLEAR:LBY <ADR.
0200 34E1 5F CLRB
0201 34E4 6F80 CLR ,Y+
0202 34E7 5A DECB
0203 34EA 26FB BNE -5
0204 34ED 17FEED LBSR DUMP0
0205 34F0 2086 BRA CHOME
0206 34F3 0703 MC80:STB <YORI
0207 34F6 170088 LBSR SINIT
0208 34F9 3501 BITA #1
0209 34FC 2670 BNE INSG
020A 34FF 8880 EDRA ##80
020B 34FF 2B13 BMI MC81
020C 34FF 970E STA <REE
020D 34FF 9603 LDA <YORI
020E 34FF C610 LDB ##10
020F 34FF 3D MUL
0210 34FF A684 LDA ,X
0211 34FF 349F ANDA ##F
0212 34FF A784 STA ,X
0213 34FF C684 ORB ,X
0214 34FF F784 STB ,X
0215 34FF 2080 BRA DCGM
0216 34FF A684 MCG1:LDA ,X
0217 34FF 349F ANDA ##F

```

## MEMORY CHANGER アセンブル・リスト

```

0218 34EE 9A03      ORA  Y0B1
0219 34F0 A7B4      STA  X
021A 34F2 8D46      MCG2 :BSR  DCGM
021B 34F4 9A0E      LDA  REG
021C 34F6 8508      BITA  #8
021D 34F8 2706      BEQ  6
021E 34FA D606      LDB  <WADR.
021F 34FC 53        COMR
0220 34FD 2601      BNE  1
0221 34FF 39        FTS
0222 3500 16FF60     LBR4 CURSR
0223 3503 4D        IN50 :TSTA
0224 3504 2A09      BPL  7
0225 3506 9603      LDA  Y0B1
0226 3508 C610      LDB  #*10
0227 350A 3D        MUL
0228 350B D703      STB  Y0B1
0229 350D 2015      BRA  IN51
022A 350F A684      LDA  X
022B 3511 1F87      TFR  A,B
022C 3513 84F0      ANDA  #*F0
022D 3515 9A03      ORA  Y0B1
022E 3517 A780      STA  X+
022F 3519 B610      LDA  #*10
0230 351B 3D        MUL
0231 351C D703      STB  Y0B1
0232 351E 1F10      TFR  X,D
0233 3520 C40F      ANDB  #*F
0234 3522 27CE      BEQ  MCG2
0235 3524 A684      IN51 :LDA  X
0236 3526 C610      LDB  #*10
0237 3528 3D        MUL
0238 3529 9A03      ORA  Y0B1
0239 352B A780      STA  X+
023A 352D D703      STB  Y0B1
023B 352F 1F10      TFR  X,D
023C 3531 C40F      ANDB  #*F
023D 3533 26EF      BNE  IN51
023E 3535 8D03      BSR  DCGM
023F 3537 16FF29     LBR4 CURSR
0240 353A 9606      DCGM :LDA  WADR.
0241 353C C610      LDB  #*10
0242 353E 3D        MUL
0243 353F 1E89      EXG  A,B
0244 3541 4F        CLRA
0245 3542 CB05      ADDE  #5
0246 3544 17FE86     LBSR LOC.
0247 3547 9E07      LDY  <ADR.
0248 3549 D606      LDB  <WADR.
0249 354B C4F0      ANDB  #*F0
024A 354D 3A        ABX
024B 354E 17FEDEF     LBSR LDISP0
024C 3551 17FE19     LBSR SDISP0
024D 3554 16FE95     LBR4 SETCUR
024E 3557 9E07      SINIT :LDY  <ADR.
024F 3559 D606      LDR  <WADR.
0250 355B C4F0      ANDB  #*F0
0251 355D 3A        ABX
0252 355E 71BDFAAE    LEAY SUM1,PC
0253 3562 C610      LDA  #*10
0254 3564 A6A4      LDA  X
0255 3566 A0B0      SUBA  X
0256 3568 A7A0      STA  X+
0257 356A 5A        DECB
0258 356B 2AF7      BNE  -9
0259 356D 9E07      LDY  <ADR.
025A 356F D606      LDB  WADR.

```

```

025B 3571 3A        ABX
025C 3572 960E      LDA  <REG
025D 3574 39        RTS
025E 3575 BDE0      ERASE0:BSR  SINIT
025F 3577 2B0C      DMI  ERASE1
0260 3579 A6B4      LDA  X
0261 357B 84F0      ANDA  #*F0
0262 357D A780      STA  X+
0263 357F 1F10      TFR  X,D
0264 3581 C40F      ANDB  #*F
0265 3583 27B5      BEQ  DCGM
0266 3585 6F80      ERASE1:CLR  X+
0267 3587 20F6      BRA  -10
0268 3589 BDCC      DUP0:BSR  SINIT
0269 358B 2B14      RMI  DUP1
026A 358D A610      LDA  -16,X
026B 358F 840F      ANDA  #*F
026C 3591 E684      LDB  X
026D 3593 C4F0      ANDB  #*F0
026E 3595 E784      STB  X
026F 3597 AAB4      ORA  X
0270 3599 A780      STA  X+
0271 359B 1F10      TFR  X,D
0272 359D C40F      ANDB  #*F
0273 359F 2799      BEQ  DCGM
0274 35A1 A610      DUP1 :LDA  -16,X
0275 35A3 A780      STA  X+
0276 35A5 20F4      BRA  -12
0277 35A7 8DAE      DEL0:BSR  SINIT
0278 35A9 2A07      RPL  DEL1
0279 35AB A684      LDA  X
027A 35AD C610      LDB  #*10
027B 35AF 3D        MUL
027C 35B0 E784      STB  X
027D 35B2 A684      DEL1 :LDA  X
027E 35B4 84F0      ANDA  #*F0
027F 35B6 A784      STA  X
0280 35B8 1F10      DEL2 :TFR  X,D
0281 35BA C40F      ANDB  #*F
0282 35BC C10F      CMFB  #*F
0283 35BE 102FF7B     LBEO DCGM
0284 35C2 A601      LDA  1,X
0285 35C4 C610      LDB  #*10
0286 35C6 3D        MUL
0287 35C7 AAB4      ORA  X
0288 35C9 A780      STA  X+
0289 35CB E784      STE  X
028A 35CD 20E9      BRA  DEL2
028B 35CF DD86      BACK50:BSR  SINIT
028C 35D1 2B0D      BMI  BACK51
028D 35D3 8B80      EDRA  #*B0
028E 35D5 970E      STA  /REG
028F 35D7 A684      LDA  X
0290 35D9 C610      LDB  #*10
0291 35DB 3D        MUL
0292 35DD E784      STB  X
0293 35DE 20D8      BRA  DEL2
0294 35E0 D606      BACK51:LDB  WADR.
0295 35E2 C40F      ANDB  #*F
0296 35E4 270A      BEQ  BACK52
0297 35E6 9A06      DEC  WADR.
0298 35E8 3B80      EDRA  #*B1
0299 35EA 970E      STA  REG
029A 35EC 201F      LEAX  -1,X
029B 35EE 20C2      ERA  DEL1
029C 35F0 16FF47     BACK52:LBR4 DCGM
029D 35F2 39        EXEND :RTS

```

## MEMORY CHANGERのFM-7の変更箇所

■藤原 和典

実はサブCPUの方のROMを使っていてそこで暴走します。変更は\$310Cから22バイトをリスト1のように変更します。すこし手抜きをしていますがこれぐらいの変更で動きます。\$3075番地からリスト2のようにすれば、  
 “\*”, “/”, “+”, “-”, “=”, “”, “” がA~Fとなります。

## リスト2

```

Add  +0 +1 +2 +3 +4 +5 +6 +7
3075 2A 2F 2B 2D 3D 2C 49 02

```

## リスト1

```

B0FD9E      KEYIN: JSR  $FD9E
8637        LDA  #*37
B0EDCB      JSR  $EDCB
B0FD6E      JSR  $FD6E
FDD383      STD  $D383
8112        CMPA #*12
2621        BNE  $3121
39          RTS

```



# エディタ・アセンブラ

## V.2.0

■馬場二郎

'82年5月号の発表後、モニタを自作してオールRAMで動作するようにして使ってきました。最近、大きなプログラムを作っていると、どうしてもエディタ関係が使いづらくいられます。

たとえば、後でコメントを加えたいとかラベルを加えたいとか1文字間違えていたなど、初めから入力せねばならず、へたをすると関係のない行まで削除しかねません。

FM-7も出たことなので、この際両者共通に使えるバージョンとして機能アップしました。操作は前回のバージョンをほぼ完全に受け継ぐようにしました。改善点は主に次のようになります。

- ①オールRAM動作可能
- ②スクリーン・エディットによる変更可能
- ③ラベルとステートメントNo.のクロス・リファレンス出力
- ④サーチ、コピーなどのコマンド追加

### テキストの入力とアセンブル

\$4000からスタートさせると、システムの状態が表示されます。初めはSコマンドでテキストをクリアしてください。EまたはI0コマンドでインサート・モードになります。

例 

まずORGを指定しないと0番地からアセンブルするので、BASICに戻ると暴走したりするので、注意してください。

また、テキスト入力後にエラーが表示されると追加されません。再度入力直してください。

アセンブルはまずコマンド"1"でラベルを登録し、コマンド"2"で実行します。

### スクリーン・エディット

本プログラムのテキストはBASICテキストは根本的に異なっているので、同様な操作はできません。原則として

#### 現在のメモリ・ロケーション

ワーク・エリア	0000
テキスト領域	2000
エディタ・アセンブラ	4000
ラベル	5BFF
システム・スタック	↓
ROM	8000
	↑

次のように行なう必要があります。

- ①スクリーン・エディットはコマンド入力モード(>表示)でしかできず、インサート時は行なえない。
- ②行Noを変えることはできない。また、誤って変更したときは行Noを戻すか[CLEAR]キーを押す。
- ③対象はLコマンドのみで、しかも直後でなければならない。
- ④以下で説明するI、Dコマンドでの表示については再度の変更、削除を行なうことはできない。

このようにしなければならない理由は、テキストが絶対行Noを持たず、全体からの相対位置でとらえられるので、削除、追加などによりNoが常に変わる可能性があるからです。したがって、この点に注意してもらえれば②③④の条件は無視してもかまいません。

☐ ×××× TEXT: NOP; コメント  
☐ (ブランク) でテキストを変更→変更  
 D →削除  
 I →インサート行の指定

変更、削除とも複数行を同時に行なえます。Iコマンドのときは最上段のみが有効になりますが、変更は同時にはできません。また入力後エラーメッセージが出たものは、テキストの訂正は行なわれません。

### 二モニック

命令はほとんどモトローラ仕様になっていると思いますが、擬似命令やラベルの使用はかなり使い方に制限があり



ます。

#### ▶擬似命令

FCB…1 バイト定数。

= (EQU)…ラベルの登録。

FDB…2 バイト定数。

RMB…2 バイト定数。領域の割り付け。

FCC…文字データ、スラッシュ(/)で囲む。

#### ▶定数 10進…符号なしまたは“-”

16進…“\$”

文字…“,” で1バイトのみ修飾

#### ▶その他 ANDCC→ANDC(4バイトに統一のため)。

BLO→BCS, BHS→BCC,

特殊レジスタは、ダイレクト・ページ(DP)。

コンディション・コード(CC)、プログラム

・カウンタ(PC)です。

## ラベル、定数の演算

＋の演算のみ許します(ラベル演算には先にラベルがあることが必要。1 + LABELは許されない)。

FCB…(10進, 16進, 文字定数の演算)。

EQU…( “ ” ), “\*” で現在のアドレス。

RMB…( “ ” )。

FDB…(ラベル, 16進, 10進, 文字定数の演算)。

・部制限があります。例1を見てください。

## オフセット

ラベルが使えると同時に演算も行なえるため、次のようにモードを決めます。演算中1つでも10進以外があれば、すべてHEX形式のオフセットとみなします(例2)。

HEX形式…すべて16ビット・オフセット、8ビットにするには“<”で強制的に行なう。5ビット・オフセットの指定はできない。

10進形式…符号付きと見て、自動的に5ビット、8ビット、16ビット・モードを決める。

## ロケーションの変更とオールRAM動作

\$××00単位でポジション・インディペンデントですから、どこに配置してもかまいません。このとき、\$4007～8はラベル・エリアの先頭アドレス、\$4009～Aテキストの先頭アドレスを必要に応じて変更してください。

オールRAM動作を実質的に行なうためには、この他にテープなどに入・出力の行なえる機能をもつモニタが必要です。FM-7はソフトで切り替えられますが、やはりROMを使うことはできず同様です。

FM-8ではモニタとアセンブラ、テキストをすべて裏RAMで使うことをおすすめします。ディップ・スイッチでプロテクトをかけてオブジェクトをテストすることにより、かなり効率的にデバッグを行なえます。

B……エディタ・アセンブラから抜ける(RTS)。

C×××××<sup>(1)</sup>×××××<sup>(2)</sup>×××××<sup>(3)</sup>……行No.(1)から(2)までを(3)の後ろへコピー(ただし、(1)<(3)<(2))。

D×××××<sup>(1)</sup>[ , ×××××<sup>(2)</sup> ]……行No.(1) [ から(2)まで ] を削除。

E……テキストの最後からインサート・モードへ。

G×××××<sup>(1)</sup>×××××<sup>(2)</sup>×××××<sup>(3)</sup>……行No.(1)から(2)を(3)の後ろへ移動する。

H……コマンド・リスト表示。

I×××××<sup>(1)</sup>……(1)の後ろよりインサートモードへ。

L[×××××]……テキスト・リスト表示。パラメータ省略時は現在のポインタからリスト。

M×××××<sup>(2)</sup>……行No.(1)を削除し、その行からインサート。

P……“2”および“X”実行時にプリンタ出力するためのスイッチのON/OFF。

PO[ ( , ×××××<sup>(1)</sup> ) ]……プリンタ・イニシャライズ(例、MP-80TYPE II, PO, \$1B, 'A', 9で改行を9ピッチに)。

Q “string”……ポインタ行より文字列をサーチする。見つければその点へポインタが移る。

R……テキストやラベルの使用状態などを見る。

S……テキスト領域のクリア(誤って入力したらテキスト・エリアの先頭データ“00”を文字に変える)。

T……“2”実行時に画面リストを表示するかのスイッチ ON/OFF。

U……ポインタを0に。

W{ 80 } , { 25 } , { 40 } , { 20 }……画面サイズ指定。例 W40, 25

X……クロス・リファレンス出力 (“1”実行後に行なうこと)。

1……パス1(ラベル登録)。

2……パス2 アセンブルを行ない、オブジェクトを出力 (“1”を行なっていること)。

×××××は10進数字

“Q” コマンドを追加したこともありポインタを採用しました。ポインタを変えることのできるコマンドは“L”のパラメータと“Q”, “U”だけです。

“L” “X” “2” を止めるときは **SPACE** キーを、再開は **SPACE** キーで、止めるなら他のキーを押します。

### エディタ・アセンブラ マシン語ダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4000	34	7F	1C	00	16	01	F9	5C	00	10	00	00	00	00	00	00	14E
4010	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
4020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
4030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
4040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
4050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
4060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
4070	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

40B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
40C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
40D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
40E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
40F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

## エディタ・アセンブラ マシン語ダンプ・リスト

Sum:	34	7F	1C	00	16	01	F9	5C	00	10	00	00	00	00	00	00	:4B
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4100	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4110	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4120	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4130	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4140	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4150	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4160	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4170	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4180	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4190	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
41A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
41B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
41C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
41D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:0A
41E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
41F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
Sum:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4200	30	BD	FE	07	1F	10	1F	B8	17	01	39	0F	21	0F	14	03	:42
4210	14	0F	23	16	00	00	0F	08	0F	21	0F	23	17	01	96	17	:1D
4220	02	94	11	87	3E	00	A7	01	32	A6	00	81	3E	10	26	0E	:FF
4230	EC	17	01	81	A6	A4	30	BC	07	33	BC	18	BD	3C	20	D6	:28
4240	4C	44	49	32	31	00	55	C2	53	52	54	0D	5B	43	47	51	:55
4250	45	57	48	00	02	9F	05	CA	04	58	07	2E	06	72	00	4B	:A5
4260	00	3E	00	3E	00	73	00	71	05	03	0F	A2	10	F2	11	63	:5D
4270	11	8C	12	1C	12	83	12	56	12	BE	5F	6D	85	27	10	A1	:93
4280	85	27	03	5C	20	F5	32	62	58	EC	C5	33	CB	1F	35	39	:4B
4290	35	FF	6F	9D	FD	73	4F	5D	DF	47	20	2B	A6	21	87	47	:64
42A0	27	04	03	13	20	21	31	22	A6	00	B1	0D	10	27	FF	6E	:4E
42B0	81	2C	26	0E	17	08	B2	26	09	1F	10	1F	9B	17	01	DC	:85
42C0	20	E6	16	01	FE	03	14	17	01	EC	0D	2A	2A	2A	20	36	:17
42D0	3B	30	39	20	41	53	53	45	4D	42	4C	45	52	20	56	65	:3A
42E0	72	20	32	2E	30	20	2A	2A	2A	0D	54	45	5B	54	41	3A	:4C
42F0	17	03	94	9E	17	17	02	02	17	01	B8	20	4C	41	42	45	:6B
Sum:	17	3B	B6	B8	22	34	95	7E	3F	36	14	BE	1A	BB	0A	47	:3E
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4300	4C	3A	00	9E	07	6D	B4	27	04	30	08	20	F8	17	02	0D	:80
4310	17	01	A3	20	50	52	49	4E	54	45	52	3A	00	96	13	BD	:6F
4320	11	E4	01	92	00	4C	49	53	54	3A	00	96	14	BD	03	16	:8E
4330	FE	17	4D	27	07	17	01	7E	4F	4E	00	39	17	01	07	47	:A7
4340	46	46	40	39	34	06	17	00	EA	EC	03	FD	FC	B2	17	16	:D1
4350	00	F8	35	B6	34	56	30	BC	4E	20	05	34	56	30	B2	45	:A0
4360	17	00	00	CE	FC	80	6D	FC	4F	41	EC	84	ED	42	17	65	:C4
4370	DC	17	00	87	30	80	BF	DC	AE	A3	81	02	27	47	E6	44	:00
4380	27	13	C0	03	26	07	33	48	A1	50	25	02	C6	17	A6	C0	:61
4390	A7	B0	5A	08	F9	86	0D	A7	B4	17	0D	B9	17	F8	AE	31	:24
43A0	BD	FD	A3	35	D6	04	00	05	00	BD	04	5A	24	00	4D	2C	:58
43B0	02	86	20	20	04	34	02	86	0D	BD	02	35	82	0D	4D	25	:1B
43C0	31	81	11	26	06	0C	4D	0C	2D	02	81	12	26	04	0C	01	:B1
43D0	4D	20	F2	81	13	27	F8	B1	0D	26	09	8D	07	86	0A	8D	:80
43E0	03	86	0D	39	0D	23	26	02	BD	11	0D	21	27	03	17	00	:34
43F0	AB	39	0A	4D	0D	23	26	02	BD	01	39	34	56	CE	FC	80	:2E
Sum:	34	04	ED	66	2C	D1	9D	6F	0E	46	79	93	A5	94	95	C6	:8B
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4400	30	BC	25	E6	8C	21	A7	95	5C	E7	8C	1B	0D	4D	26	12	:1C
4410	BD	21	C6	07	30	1C	A6	80	E7	00	5A	26	F9	8D	2E	6F	:F7
4420	8C	05	35	C6	00	00	03	00	00	00	00	34	04	C6	18	5A	:0F
4430	26	FD	BC	34	04	0F	FD	00	2B	F8	1A	50	C6	80	F7	FD	:A9
4440	05	C6	08	5A	27	05	FD	00	00	00	00	00	C6	80	39	34	:14
4450	8D	02	BD	00	39	F6	FC	B2	8D	E3	8D	BF	B6	FC	83	8D	:0C
4460	8D	01	CA	27	00	FD	FC	82	8D	E3	8D	ED	02	E7	05	17	:05
4470	E4	BD	DC	4D	35	94	34	56	30	8C	1B	5A	26	F9	17	FF	:E7
4480	FF	B1	CE	FC	B2	C6	06	A6	80	07	C0	00	34	04	1C	AF	:14
4490	8C	35	D6	01	00	50	19	00	19	00	00	00	34	04	1C	AF	:14
44A0	F6	FD	02	54	25	F4	BF	FD	01	5F	E7	FD	00	CA	04	7F	:71
44B0	FD	00	1A	50	35	84	AE	E1	A6	80	27	05	17	FE	FE	20	:34
44C0	17	F1	45	8D	F1	11	87	07	46	4D	44	45	52	52	0D	00	:12
44D0	16	FD	A3	8D	F1	11	87	07	46	4D	44	45	52	52	0D	00	:40
44E0	86	01	39	8D	D1	11	87	07	5E	47	4D	45	52	52	0D	00	:97
44F0	B6	01	39	A6	21	81	0D	27	0E	17	09	73	26	C5	8C	00	:54
Sum:	39	D6	71	B5	F5	07	1F	1F	E1	39	61	40	19	3B	40	5A	:1B
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4500	00	27	04	30	1F	20	02	9E	47	9F	15	17	01	B2	26	B3	:AA
4510	17	FE	A2	9F	47	10	9E	19	A6	A4	10	27	FC	FB	8D	4F	:B5
4520	34	10	17	FF	91	11	87	20	00	35	10	17	03	F5	30	01	:92
4530	17	0F	44	17	FE	79	17	0B	41	A6	A4	81	3B	27	27	C6	:75
4540	07	17	00	4B	C6	05	17	00	41	A6	81	3B	27	07	B1	0D	:38
4550	18	C6	0F	17	00	39	81	0D	27	0C	20	05	C6	17	FE	10	:11
4560	48	C6	C6	3B	17	FE	57	17	00	1F	0B	64	16	FF	A9	17	:43
4570	FE	EC	16	20	26	0E	17	FE	E5	23	FB	81	20	27	05	35	:1D
4580	06	16	FC	92	39	A6	A0	17	FE	33	81	0D	26	F7	39	A6	:FB
4590	A0	97	36	81	3B	26	02	86	20	17	FE	21	8D	20	0D	37	:1E
45A0	27	04	0A	37	20	05	17	00	BE	27	06	5A	26	E1	5C	20	:7A
45B0	DE	81	0D	27	06	5A	27	03	17	FD	EE	96	36	39	81	27	:0C
45C0	27	05	81	2F	27	0D	39	0D	37	26	FB	34	02	86	02	97	:33
45D0	37	35	82	34	02	0E	37	26	04	86	46	20	F2	4F	20	EF	:CE
45E0	34	06	1F	10	8D	06	1E	89	8D	02	35	86	34	06	1F	89	:1F
45F0	B4	F0	44	44	44	44	8D	08	1E	89	B4	0F	8D	02	35	86	:9D
Sum:	8B	35	7D	A6	73	EC	FF	51	75	91	A7	EE	0A	C4	75	F3	:60
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4600	BB	30	81	B3	24	25	02	BB	07	17	FD	B2	39	34	02	C6	:10

4610	17	FF	A2	35	02	16	FF	4E	34	20	31	21	BE	00	00	A6	:2C	
4620	A0	B0	30	25	55	B1	0A	25	06	80	07	B1	10	24	4B	34	:3B	
4630	06	1F	10	58	49	5B	49	5B	49	5B	49	1F	01	35	06	1E	:32	
4640	89	3A	A6	14	17	00	10	26	D6	20	3A	B1	23	27	2A	B1	:FA	
4650	3C	27	26	B1	5B	27	22	B1	C2	27	1E	B1	5D	27	1A	B1	:40	
4660	2B	27	16	B1	2D	27	12	B1	20	27	0E	B1	0D	27	0A	B1	:65	
4670	3A	27	26	B1	3B	27	02	B1	3D	39	20	86	01	39	35	B8	:8D	
4680	06	B6	01	97	12	4F	39	9E	09	A6	B0	26	FC	9F	17	39	:9C	
4690	BE	00	00	DE	09	9C	15	27	0C	A6	C0	27	0C	B1	0D	26	:A6	
46A0	FB	30	01	20	F0	DF	19	4F	39	B6	01	39	17	0C	26	7D	:7D	
46B0	21	9F	15	17	FF	DA	26	1A	17	00	B7	27	05	17	FE	13	:F7	
46C0	20	F6	17	02	D7	27	05	17	FE	19	20	EC	B0	07	26	02	:2B	
46D0	20	E6	16	FD	EE	17	FF	AF	17	FF	B5	26	26	17	00	26	:20	
46E0	9A	19	10	9E	17	DC	19	D3	18	2A	03	17	00	2C	30	B0	:B1	
46F0	FA	14	10	9E	1B	DE	19	17	00	1F	DC	15	C3	00	01	DD	:31	
Sum1	F7	6B	AF	FA	A0	02	E6	59	BE	CF	44	BB	80	59	D7	EA	:AF	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
4700	15	4F	39	B6	01	39	30	8D	FA	BC	CE	00	01	A6	B0	B1	:16	
4710	0D	27	04	33	41	20	F6	DF	1B	39	4F	7E	1F	10	A3	7E	:82	
4720	1F	02	31	21	E1	7E	4C	7E	24	0F	1F	20	30	BB	33	C8	:35	
4730	A6	B2	A7	C2	31	3F	26	FB	39	A6	B0	C7	01	31	3F	26	:7B	
4740	FB	39	17	FD	71	11	B7	00	17	FC	10	17	FC	67	30	B0	:AB	
4750	FA	44	17	01	2D	B1	0D	10	27	00	A7	B1	3B	10	27	00	:E2	
4760	BC	3A	20	A6	A0	17	FE	FF	26	F9	35	20	B1	3A	27	0A	:9A	
4770	B1	3D	27	06	B6	20	A7	80	20	2B	C6	07	A6	A4	A7	B0	:41	
4780	B1	3A	27	1B	B1	3D	27	07	C1	21	5A	27	7A	20	ED	C1	:01	
4790	06	24	74	31	21	17	00	EA	C6	1F	20	37	C1	06	24	67	:7F	
47A0	31	21	17	00	DD	C6	05	A6	A4	17	FE	B6	27	0B	A7	B4	:8B	
47B0	31	21	30	01	5A	27	50	20	EE	B1	04	25	4A	17	0C	C2	:2F	
47C0	B1	3D	27	35	B1	3B	27	25	A6	A4	17	FE	9A	27	3B	B6	:00	
47D0	20	A7	B0	A6	A4	B1	2F	27	C1	17	FE	BB	27	0B	A7	B4	:93	
47E0	31	21	30	04	20	ED	17	00	99	B1	0D	27	0C	B6	3B	A7	:69	
47F0	B0	31	21	A6	A4	B1	0D	26	F6	B6	0D	A7	80	31	21	B6	:5B	
Sum1	21	BE	64	12	EB	4A	27	9A	E5	34	79	99	67	05	AD	AC	:0B	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
4800	00	39	35	07	13	1A	FA	0F	B6	01	39	A7	B0	31	21	A6	A4	:13
4810	B1	0D	27	F3	B1	2F	26	F2	A7	B0	31	21	20	C8	B0	1F	:7D	
4820	10	26	FC	9F	16	F9	FC	34	20	B0	14	35	20	10	26	FC	:AB	
4830	92	17	06	3B	34	26	FC	B8	30	1F	9F	15	16	FE	17	14	:AE	
4840	06	2D	26	3B	A4	10	3D	17	9F	15	17	FE	43	35	10	26	:9E	
4850	2E	DE	19	A6	A4	20	00	27	05	17	06	13	26	21	9F	15	:5A	
4860	34	40	17	FE	2B	35	40	26	16	11	93	19	2A	11	9E	19	:14	
4870	34	10	17	FE	12	35	10	10	9E	17	17	FE	9D	4F	39	B6	:35	
4880	01	39	A6	A4	B1	20	26	04	C1	21	20	F6	39	96	0B	0F	:A0	
4890	23	36	32	27	01	39	34	30	0D	14	26	04	B6	01	97	23	:2D	
48A0	17	0D	F0	10	26	00	0B	0D	2C	27	02	0F	03	25	B0	0F	:A7	
48B0	23	35	30	35	06	14	10	26	F9	5B	30	09	17	10	B7	B7	:78	
48C0	17	FC	20	16	F9	50	B6	31	97	0B	10	9E	09	9E	07	4F	:B6	
48D0	B4	BE	00	00	9F	0C	9E	0C	9F	0E	A4	30	17	00	C1	35	:B5	
48E0	30	27	06	17	FB	FD	16	F9	2D	A6	A4	B1	20	27	07	B1	:42	
48F0	3B	27	03	17	00	39	B0	0D	A6	A0	B1	0D	26	FA	A6	A4	:BD	
Sum1	23	A5	F9	01	13	3B	EA	47	E9	6D	5E	AB	00	4F	2A	71	:B7	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
4900	26	D4	10	F6	F9	11	34	70	DE	0C	0D	29	27	0A	1F	30	:D3	:31
4910	2A	1F	03	DF	0C	35	F0	30	8D	F7	9F	9C	32	27	FA	96	:2E	
4920	0B	B1	32	26	A6	B4	A7	C4	30	01	33	41	20	EC	34	16	:12	
4930	20	17	06	04	35	20	26	1B	34	A0	A0	17	FD	2B	27	01	:D1	
4940	05	17	FA	79	20	F4	B6	0D	17	FA	72	35	20	B6	01	39	:CE	
4950	34	20	9E	07	C6	0B	A6	B4	27	03	30	20	F9	C6	06	A6	:E0	
4960	A4	B1	3A	27	0B	B1	3D	27	07	A7	80	31	21	5A	2C	EF	:6B	
4970	5D	27	05	0F	80	5A	20	F8	DE	0E	EF	B1	6F	B4	35	20	:BE	
4980	4F	39	B6	32	97	0B	8E	00	00	9F	0C	9E	13	27	02	97	:B4	
4990	21	10	9E	09	8E	00	00	9F	15	16	FB	7C	31	BD	F7	F6	:52	
49A0	30	BD	F7	0E	0F	2B	0F	2C	0F	29	30	BD	F7	0C	9F	32	:05	
49B0	A6	A4	B1	3B	27	1F	B1	3D	10	27	00	96	17	FC	9B	27	:A9	
49C0	04	31	21	20	EB	31	21	17	05	50	27	0A	17	05	5A	10	:D6	
49D0	27	06	22	B6	01	39	1F	13	A6	C4	B1	FF	10	27	00	B7	:E9	
49E0	B1	FE	10	17	20	0E	B1	FD	10	27	00	B6	B1	FC	10	27	:67	
49F0	00	DD	3C	B1	FB	27	5C	B1	FA	10	27	00	9B	17	01	9F	A6	:7C
Sum1	A7	EC	9B	6C	35	AC	F3	A6	B3	6D	69	30	4E	72	D9	FC	:5F	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
4A00	C4	B4	10	10	26	00	F7	A6	A4	B1	0D	10	27	00	D9	B1	:EE	
4A10	3B	10	27	0D	D3	31	21	34	20	17	01	72	35	20	B1	2C	:77	
4A20	10	27	02	1B	A6	A4	B1	23	10	27	01	B8	B1	3C	10	27	:F6	
4A30	01	C5	B1	5B	10	27	01	DE	A6	C4	B4	02	27	95	A6	43	:4D	
4A40	17	01	23	17	03	9C	1F	10	17	01	20	1F	9B	17	01	1B	:42	
4A50	4F	39	31	21	A6	A4	B1	2A	26	01	39	17	03	DB	26	04	:4E	
4A60	9F	0E	4F	39	B6	01	39	31	21	17	03	CD	26	F6	9F	0C	:F5	
4A70	9F	0E	4F	39	C6	46	10	21	34	04	17	03	BC	35	04	26	:00	
4A80	E3	34	04	1F	10	1F	9B	35	04	17	00	DA	A6	A4	5A	27	:F6	
4A90	04	B1	2C	27	E1	4F	39	B6	01	97	29	31	21	17	03	99	:B7	
4AA0	26	59	9F	2A	4F	39	C6	46	31	21	A6	A0	B1	2F	26	4B	:95	
4AB0	A6	A0	B1	2F	27	0D	5D	27	42	B1	0F	2D	3E	17	00	A6	:A0	
4AC0	5A	20	ED	4F	39	C6	04	31	21	0F	2B	5D	27	2D	34	04	:31	
4AD0	17	03	0F	1F	10	17	00	93	1F	9B	17	00	BE	A6	A4	35	:DD	
4AE0	04	5A	B1	2C	27	E1	4F	39	B6	C4	B4	20	27	0D	A6	C4	:47	
4AF0	84	10	26	07	A6	41	17	00	6D	4F	39	B6	01	39	0F	1F	:A2	
Sum1	60	11	9F	70	21	36	0B	BC	D7	AA	DE	E7	E4	2B	EA	35	:DC	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
4B00	A6	14	17	00	61	B1	1F	27	22	B1	1E	27	1E	31	21	17	:95	
4B10	04	33	26	E7	A6	B1	FF	27	E1	9A	1F	97	1F	A6	A4	:AF		
4B20	B1	2C	27	E9	96	1F	17	00	3D	4F	39	31	21	17	04	15	:D0	
4B30	26	C9	A6	01	48	4B	48	4B	97	1F	A6	A4	B1	2C	26	BB	:44	
4B40	31	21	17	04	00	26	B4	A6	01	9A	1F	1F	B9	17	00	16	:7C	

## エディタ・アセンブラ マシン語ダンブ・リスト

4B50	1F	98	84	80	27	08	1F	98	84	08	27	9F	4F	39	1F	98	:32
4B60	84	08	26	97	4F	39	34	12	4F	20	04	34	12	96	28	9E	:2C
4B70	32	A7	88	46	35	12	20	0C	34	04	D6	20	35	04	27	E6	:8E
4B80	8A	10	20	E2	34	10	9E	32	A7	80	9F	32	35	90	A6	A4	:B7
4B90	17	FA	D4	27	08	B1	2C	27	04	31	21	20	F1	39	A6	C4	:F2
4BA0	84	80	27	05	86	11	8D	BE	39	A6	C4	84	40	27	04	86	:2A
4BB0	10	8D	B3	39	A6	C4	84	08	26	09	A6	C4	84	04	26	1D	:E3
4BC0	86	01	39	A6	41	17	FF	9E	31	21	17	02	15	10	26	00	:11
4BD0	09	1F	10	1F	98	17	FF	93	4F	39	86	01	39	A6	41	17	:DE
4BE0	FF	84	31	21	17	01	F8	10	26	FF	EF	1F	10	17	FF	7B	:CC
4BF0	1F	98	17	FF	76	4F	39	A6	C4	84	02	10	27	00	90	A6	:2B
Sum:	39	24	B2	5E	5E	C9	33	D0	99	D3	6F	F9	E5	3E	CB	00	:59
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4C00	42	17	FF	62	31	21	17	01	D9	10	26	00	B2	1F	10	1F	:03
4C10	98	17	FF	57	4F	39	31	21	A6	C4	84	01	27	71	A6	44	:50
4C20	17	FF	43	86	FF	17	FF	3E	17	01	B7	26	62	A6	A4	81	:54
4C30	5D	26	5C	1F	10	17	FF	33	1F	98	17	FF	2E	4F	39	0F	:E9
4C40	20	A6	C4	84	01	27	48	A6	44	17	FF	1A	A6	A4	81	2C	:8F
4C50	10	27	01	1B	81	5B	26	08	17	00	2D	A6	A4	81	2C	10	:AB
4C60	27	01	0C	17	02	DF	26	2A	A6	02	B1	FF	27	21	97	1F	:A2
4C70	31	21	17	02	0D	26	18	A6	03	B1	FF	27	12	81	BC	27	:0F
4C80	0E	9A	1F	17	FE	F2	4F	39	86	01	97	20	31	21	39	86	:A5
4C90	01	39	34	20	0F	22	17	FE	F5	31	21	17	02	A7	35	20	:30
4CA0	26	ED	A6	03	B1	8C	27	5B	81	FF	27	E3	97	1F	17	01	:A3
4CB0	88	27	0F	A6	A4	81	3C	26	04	97	22	31	21	17	01	22	:34
4CC0	26	CD	17	02	0D	96	25	B1	02	27	25	B1	01	27	11	96	:B6
4CD0	20	26	0D	1F	10	C4	1F	DA	1F	1F	98	17	FE	88	4F	39	:3A
4CE0	96	1F	BA	88	17	FE	91	1F	10	1F	98	17	FE	78	4F	39	:68
4CF0	96	1F	BA	89	17	FE	B1	1F	10	17	FE	6A	1F	98	17	FE	:DB
Sum:	05	5A	C5	2B	23	86	11	65	FA	4B	7B	70	C3	09	AF	44	:57
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4D00	65	4F	39	0F	22	0F	24	A6	A4	81	3C	26	04	97	24	31	:6E
4D10	21	17	01	25	27	4A	17	02	BC	30	1E	4D	27	03	17	03	:B3
4D20	52	A6	C4	84	00	27	02	30	1F	0F	12	17	02	67	96	24	:D3
4D30	81	3C	27	13	30	1E	86	BD	17	FE	3D	1F	10	17	FE	2B	:19
4D40	1F	98	17	FE	26	4F	39	86	25	30	1F	B1	02	25	03	17	:46
4D50	03	21	B6	8C	17	FE	21	1F	10	1F	98	17	FE	0D	4F	39	:FC
4D60	17	02	32	0D	26	4B	96	25	B1	02	24	C9	20	E3	31	:EC	
4D70	21	A6	A4	81	2D	27	3E	17	01	CB	26	48	A6	03	B1	FF	:FB
4D80	27	45	B1	8C	27	41	97	1F	A6	A4	81	2B	27	09	96	1F	:72
4D90	8A	84	17	FD	E3	4F	39	31	21	A6	A4	81	2B	27	0D	96	:9F
4DA0	20	26	24	96	1F	8A	80	17	FD	BC	4F	39	96	1F	8A	81	:41
4DB0	17	FD	C5	4F	39	31	21	A6	A4	81	2D	27	0D	96	20	26	:BB
4DC0	06	86	82	97	1F	20	09	86	01	39	31	21	B6	83	97	1F	:BE
4DD0	17	01	72	26	F2	A6	03	B1	FF	27	EC	9A	1F	17	FD	98	:43
4DE0	4F	39	8E	00	9F	10	0F	12	8D	1C	1F	10	D3	10	DD	17	:7E
4DF0	10	0D	28	27	04	B6	01	97	2C	A6	A4	81	2B	27	EA	81	:42
Sum:	17	62	C3	35	3E	6E	D4	B1	97	73	06	17	B1	E6	60	74	:D4
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4E00	2D	27	E6	9E	10	4F	39	8E	00	0F	27	A6	A4	81	2B	:2A	
4E10	26	02	31	21	B1	2D	26	06	86	01	97	27	31	21	8D	56	:CE
4E20	27	16	96	0B	B1	32	10	27	00	C1	8E	00	00	A6	A4	17	:78
4E30	FB	25	27	04	31	21	20	F5	39	8E	00	00	9F	10	0F	27	:5B
4E40	0F	12	A6	A4	81	2B	26	02	31	21	B1	2D	26	06	31	21	:8D
4E50	86	01	97	27	BD	20	26	14	1F	10	D3	10	DD	10	A6	A4	:75
4E60	81	2B	27	DE	B1	2D	27	DA	9E	10	4F	39	86	01	39	34	:8A
4E70	20	0F	27	31	21	BC	34	20	5F	8E	00	00	A6	A4	81	27	:67
4E80	27	5B	B1	24	10	27	F7	92	80	30	25	4B	B1	0A	24	47	:FA
4E90	20	0F	A6	A4	17	F7	C0	2D	21	80	30	25	3A	B1	0A	24	:4D
4EA0	36	34	06	1F	10	58	49	DD	1D	58	49	58	49	D3	10	1F	:8B
4EB0	01	35	06	1F	B9	3A	31	21	20	D8	0D	27	27	11	1F	10	:03
4EC0	43	53	C3	00	01	1F	01	84	80	27	0C	35	06	4F	39	1F	:93
4ED0	10	84	80	26	02	20	F4	16	F7	A0	1F	10	4F	31	21	E6	:B3
4EE0	A0	1F	01	35	06	86	01	97	12	4F	39	86	01	97	12	A6	:B9
4EF0	A4	B1	2A	27	08	8D	41	27	09	86	01	97	2B	9E	0C	16	:B2
Sum:	BD	FB	00	30	C4	D5	9E	CF	7C	9B	E7	15	4E	5A	34	3A	:14
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4F00	FF	2B	A6	84	E6	01	1F	01	C4	84	02	17	27	FF	1F	1F	:10
4F10	43	53	C3	00	01	1F	01	16	FF	13	CC	00	09	DD	2D	30	:B1
4F20	8D	07	DD	C6	04	D7	26	20	2B	CC	00	06	DD	2D	30	8D	:1C
4F30	08	9E	C6	04	D7	26	20	1C	CC	00	08	DD	2D	9E	07	C6	:F5
4F40	06	D7	26	20	0F	CC	00	06	DD	2D	30	8D	0C	62	C6	02	:01
4F50	D7	26	20	00	10	9F	30	A6	84	27	34	34	30	D6	26	A6	:87
4F60	B4	27	17	A1	A4	26	09	31	21	30	01	5A	27	13	20	EF	:5C
4F70	35	30	1F	10	D3	2D	1F	01	20	DA	A6	A4	17	F6	DB	26	:03
4F80	EF	5D	27	05	5A	30	01	20	F8	35	06	35	06	4F	39	10	:29
4F90	9E	30	86	01	39	96	12	26	31	1F	10	4D	27	09	B1	FF	:B9
4FA0	27	19	86	02	97	25	39	1F	10	C4	80	26	F5	1F	10	C4	:3E
4FB0	F0	26	03	0F	25	39	86	01	97	25	39	1F	10	C4	80	27	:9C
4FC0	E1	1F	10	53	C4	F0	26	EE	20	E9	1F	10	4D	26	D3	0D	:B6
4FD0	22	27	CF	20	E1	BE	00	00	0F	2B	96	0B	81	32	26	0B	:63
4FE0	17	FD	FF	26	08	1F	10	93	0C	1F	01	4F	39	8E	00	00	:45
4FF0	B6	01	97	2B	39	1F	13	17	FB	A4	A6	41	17	FB	67	31	:FB
Sum:	B4	B7	33	F7	8D	BB	D9	2F	AB	75	1A	3B	DC	24	11	93	:CE
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5000	21	17	FE	35	26	14	A6	C4	84	02	17	05	1F	10	17	FB	:02
5010	55	1F	10	1F	98	17	FB	4E	4F	39	96	0B	81	32	26	E6	:83
5020	17	FF	B2	A6	C4	84	40	27	02	30	1F	30	1F	30	1F	A6	:B2
5030	C4	84	02	26	30	1F	10	B1	FF	27	18	4D	27	03	17	00	:1C
5040	32	1F	10	C4	80	27	03	17	00	29	1F	10	1F	98	17	FB	:02
5050	1A	4F	39	1F	10	C4	80	26	03	17	00	17	1F	10	1F	98	:52
5060	17	FB															

5080	39	34	30	17	F3	29	9E	0C	17	F5	55	17	F3	21	86	0C	:98
5090	97	26	30	8D	F0	24	C6	05	9C	32	27	20	5D	27	1D	A6	:B5
50A0	88	46	26	07	A6	80	F7	F5	43	20	0A	86	2D	17	F3	0D	:84
50B0	17	F3	0A	30	01	0A	26	0A	26	5A	20	DC	9F	3A	0D	26	:01
50C0	27	07	17	F2	EA	0A	26	20	F3	17	F8	39	3C	30	4F	39	:99
50D0	96	0B	34	10	B1	32	C2	02	35	90	9E	34	9C	32	26	02	:46
50E0	35	90	C6	0D	17	F4	CE	06	05	9E	34	9C	32	27	25	5D	:85
50F0	27	19	A6	88	46	26	07	A6	80	17	F4	F0	20	0A	86	2D	:DF
Sum:	36	BF	93	4A	CE	AD	7E	4B	DC	1F	07	50	03	0D	AA	BF	:E1
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5100	17	F2	B8	17	F2	B7	30	01	5A	20	0A	9F	3A	86	0D	17	:8B
5110	F2	AB	20	CE	9F	34	86	0D	17	F2	A2	35	90	33	8D	EF	:10
5120	35	97	DF	30	08	ED	EF	61	9F	49	8E	00	9F	48	31	3F	:30
5130	A6	A4	B1	20	27	26	B1	44	10	27	00	95	B1	49	10	27	:CA
5140	00	A0	B1	0D	26	3A	33	8D	EF	0C	11	93	3F	27	45	AE	:4B
5150	C1	17	03	23	17	F2	58	17	F3	89	20	EE	17	FD	10	10	:34
5160	26	F3	60	9F	15	17	F5	E6	26	20	17	F8	2F	26	1B	31	:15
5170	70	86	0D	A4	9E	15	17	F6	C4	26	0E	9E	15	30	1F	:0E	
5180	9F	15	17	F5	50	17	F1	CC	20	A6	DE	3F	9E	15	AF	C1	:EA
5190	DF	3E	20	F1	30	8D	EE	0F	DE	49	9F	49	11	93	49	27	:ED
51A0	22	AE	C3	31	E8	0E	86	0D	AF	A4	34	0A	17	F4	95	35	:B5
51B0	40	10	26	F3	0E	AE	C4	9C	4B	2E	E1	9E	48	1F	9F	:A6	
51C0	4B	20	D9	9E	48	10	27	F0	4D	9F	E5	17	F1	E7	16	F4	:8E
51D0	E0	17	FC	9B	10	26	F2	EB	1F	10	9E	49	ED	B1	9F	49	:0D
51E0	20	A3	17	FC	8A	10	26	F2	BA	1F	10	9E	48	BC	00	00	:06
51F0	26	93	DD	4B	20	BF	17	00	B1	9E	09	DE	07	0D	13	27	:FB
Sum:	BC	CF	74	35	B6	ED	AC	C4	7F	73	4E	32	48	7B	EF	9A	:D5
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5200	02	03	21	6D	C4	27	0C	17	F3	65	BD	0A	33	4B	17	F1	:13
5210	A4	20	F0	16	F0	00	34	50	34	10	AE	46	17	F3	C1	17	:58
5220	F1	BD	35	10	C6	06	17	00	41	17	F1	83	0F	38	0F	39	:01
5230	CC	00	01	DD	3A	1A	17	00	AF	25	E2	96	39	B1	0A	26	:0A
5240	17	F1	72	C6	0C	17	F1	61	0F	39	0C	39	34	10	9E	3A	:5E
5250	17	02	24	35	10	08	3B	26	08	17	F1	53	17	F1	50	20	:CB
5260	D4	B6	2A	17	F1	S7	20	F4	35	DD	00	34	0A	A6	C0	26	:02
5270	86	20	17	F1	F4	8A	5A	26	F4	35	C0	DE	07	6D	C4	27	:24
5280	DF	3C	0F	3E	33	48	6D	C4	27	1A	30	58	BD	2C	BD	45	:68
5290	2B	06	0D	3E	26	0F	20	EA	86	01	97	3E	30	18	9C	3C	:37
52A0	2B	03	20	EA	39	30	08	34	70	31	SF	33	08	17	F4	6A	:8D
52B0	35	70	1E	13	8D	15	1E	13	20	C8	34	56	30	BD	ED	BE	:53
52C0	C6	08	06	1A	AD	80	5A	26	F9	35	D6	34	56	30	BD	ED	:13
52D0	7D	1E	13	20	EB	34	06	SF	C1	06	27	09	AE	C5	A1	85	:DA
52E0	26	03	5C	20	F3	35	86	A6	80	27	58	17	F3	50	26	1D	:A2
52F0	0F	38	B1	3B	27	0F	B1	0D	26	ED	03	38	DC	3A	C3	00	:EE
Sum:	CD	5F	0E	27	D4	AD	0E	B2	AB	FD	B3	8A	F8	76	69	D3	:D3
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5300	01	DD	3A	20	E2	A6	B0	81	0D	26	FA	20	ED	30	1F	34	:7E
5310	40	C6	06	A6	B0	B7	F3	33	27	21	A1	CE	26	10	5A	26	:1C
5320	F2	A6	84	17	F3	25	27	02	20	04	1C	FE	30	C0	35	40	:0C
5330	A6	84	17	F3	16	27	B0	30	01	20	F5	30	1F	6D	C4	27	:0E
5340	E9	20	EB	1A	01	39	8D	07	10	26	F1	77	16	EE	C7	17	:9F
5350	FB	1D	26	60	30	1F	9F	15	17	F3	35	26	57	9E	19	9F	:B3
5360	41	17	FB	08	26	4E	9F	15	17	F3	25	26	47	DC	19	93	:AA
5370	41	25	41	DD	43	17	FA	F7	26	3A	9F	15	17	F3	11	26	:24
5380	33	9E	19	1F	10	93	41	2F	05	10	93	43	25	26	9F	45	:36
5390	17	F2	FA	9E	45	10	9E	17	DC	43	33	88	17	F3	7B	DC	:E3
53A0	41	10	93	45	25	02	D3	43	1F	01	10	9E	43	DE	45	17	:B1
53B0	F3	72	4F	39	86	01	39	BD	96	10	26	F1	06	DC	45	10	:2E
53C0	93	41	24	06	DC	41	D3	43	20	02	DC	41	1F	03	D3	43	:AB
53D0	1F	01	34	50	17	F2	B0	35	50	10	9E	17	17	F3	38	16	:02
53E0	EE	34	A6	0A	B1	0D	10	27	00	83	B1	22	26	F4	SF	33	:FF
53F0	8D	ED	A3	A6	A0	B1	0D	27	74	81	22	27	05	A7	C0	5C	:1E
Sum:	EA	BB	BB	09	19	2D	9A	EA	33	2B	AF	2E	27	05	A7	C0	:5E
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5400	20	F1	47	A7	C4	3D	27	65	34	04	9E	47	30	01	9F	15	:B4
5410	17	F2	7D	10	9E	19	35	03	43	8D	ED	7A	30	01	A6	A0	:24
5420	27	3A	B1	C4	27	08	B1	0D	26	FA	30	01	20	F0	34	64	:76
5430	33	A1	5A	27	0F	A6	C0	A1	A0	26	05	0A	27	06	20	F5	:72
5440	35	64	20	DA	35	44	30	1F	9F	47	30	01	17	00	28	17	:5B
5450	F0	64	20	46	4F	55	4E	44	00	16	ED	BA	17	F0	S7	4E	:59
5460	4F	54	20	46	4F	55	4E	44	21	00	16	ED	BA	1F	F0	S3	:65
5470	4F	SF	DD	47	16	ED	9F	34	36	8C	27	10	24	22	31	8C	:AA
5480	21	C6	04	86	30	34	02	1E	10	10	A3	A4	2B	06	A3	AA	:D4
5490	6C	EA	04	F5	1E	10	35	02	17	EF	22	33	22	5A	26	E3	:AB
54A0	35	B6	03	EB	00	64	00	0A	00	01	17	P9	C2	26	25	1F	:81
54B0	10	C1	50	27	00	C4	C1	2B	26	18	34	04	17	F9	B1	35	:02
54C0	26	1F	47	7E	7E	7F	C1	19	17	27	04	C1	14	26	06	17	:EF
54D0	A5	1E	AD	42	16	EF	EC	CC	27	10	DD	15	17	F1	B1	16	:9F
54E0	F1	CF	17	EF	D1	0D	3C	3C	3C	20	20	20	20	20	43	4F	:A8
54F0	4D	4D	41	4E	4A	40	40	40	43	53	54	20	20	20	20	20	:3E
Sum:	2F	3E	6F	D6	E4	23	9C	AC	42	50	DB	22	27	8E	B7	BC	:55
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5500	3E	3E	0D	42	20	13	12	A5	20	BC	AD	B3	DB	AE	B3	20	:4A
5510	28	52	54	53	29	0D	43	61	2C	62	2C	63	20	13	0D	A5	:FD
5520	20	61	20	86	07	20	62	20	A6	20	63	20	D6	08	20	BA	:A1
5530	CB	DF	B0	0D	44	61	5B	2C	62	5D	20	13	0D	A5	20	61	:B8
5540	20	5B	20	86	07	20	62	20	5D	20	A6	20	B8	B8	BC	DE	:1A
5550	AE	0D	45	20	13	12	A5	20	BB	B2	BA	DE	D6	DB	20	B2	:8F
5560	DD	BB	B0	C4	0D	47	61	2C	62	C2	63	20	13	0D	A5	20	:E3
5570	61	20	86	07	20	62	20	A6	20	63	20	C9	B1	C4	0D	CD	:11
5580	48	20	13	12	A5	20	BA	CF	DD	C4	DE	DB	DB	C4	0D	49	:09
5590	61	20	13	11	A5	20	61	20	C9	B1	C4	CD	20	B2	DD	BB	:60
55A0	B0	C4	0D	4C	5B	61	5D	20	13	0F	A5	20	5B	20	61	20	:E9
55B0	B7	DE	AE	B3	20	B6	07	20	5D	20	DB	BD	C4	0D	4D	61	:5A

## エディタ・アセンブラ マシン語ダンプ・リスト

55C0	20	13	11	A5	20	61	20	A6	20	BB	BB	BC	DE	AE	BA	DE	A3	
55D0	20	B2	DD	BB	B0	C4	0D	50	20	13	12	A5	20	CC	DF	D8	1C8	
55E0	DD	C0	B0	20	4F	AE	A5	4F	46	46	0D	50	4F	5B	5B	2C	1B	
55F0	61	5D	5D	20	13	0B	A5	20	CC	DF	DD	DD	CO	B0	20	B2	1C0	
Sum:	EB	D7	D8	8B	72	51	60	F8	56	93	0D	40	39	C7	FA	B6	126	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
5600	C6	BC	AC	D7	B2	BD	DE	0D	51	22	73	74	72	69	6E	67	169	
5610	22	20	13	0A	A5	20	D3	BC	DA	C2	20	BB	B0	C1	0D	1B6		
5620	52	20	13	12	A5	20	BC	BD	C3	D1	20	BC	DE	AE	B3	B7	13B	
5630	AE	B3	0D	53	20	13	12	A5	20	C3	87	BD	C4	20	BB	D8	176	
5640	B1	0D	54	20	13	12	A5	20	C3	87	BD	C4	20	CB	AE	B3	163	
5650	BC	DE	20	4F	AE	A5	4F	46	46	0D	55	20	13	12	A5	20	143	
5660	CE	DF	D2	DD	C0	B0	20	30	CD	0D	57	7B	38	30	2C	34	170	
5670	30	7D	2C	7B	32	35	2C	32	30	7D	20	13	03	A5	20	B6	177	
5680	DE	D2	DD	20	B7	D8	B6	B4	0D	58	20	13	12	A5	20	B8	1CD	
5690	DB	BD	DB	CC	A7	DA	DD	BD	20	D8	BD	C4	0D	31	20	13	141	
56A0	12	A5	20	D7	CD	DE	D9	20	C4	B3	DB	BD	0D	32	20	13	1CE	
56B0	12	A5	20	B1	BE	DD	CC	DE	D9	20	B8	BD	CE	AF	BA	B3	0D	17C
56C0	16	EB	53	20	FE	00	00	00	00	00	00	00	00	00	00	00	172	
56D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
56E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
56F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
Sum:	46	BA	79	A1	56	19	F7	62	E2	E1	09	EC	18	5B	4C	9E	1F7	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
5700	41	42	58	00	20	3A	00	00	00	41	44	43	41	0B	89	99	16B	
5710	B9	A9	41	44	43	42	0B	C9	D9	F9	E9	41	44	44	41	0B	110	
5720	8B	9B	BB	AB	41	44	44	42	0B	CB	DB	FB	EB	41	44	44	1F7	
5730	44	07	C3	D3	F3	E3	41	4E	44	41	0B	84	94	B4	A4	41	1B7	
5740	4E	44	42	0B	C4	D4	F4	E4	41	4E	44	43	0B	1C	00	00	189	
5750	00	41	53	4C	41	20	48	00	00	41	53	4C	42	20	58	123		
5760	00	00	00	41	53	4C	00	03	00	78	68	41	53	52	41	1F2		
5770	20	47	00	00	00	41	53	52	42	20	57	00	00	00	41	53	19A	
5780	52	00	03	00	07	77	67	42	49	54	41	0B	85	95	B5	A5	1D9	
5790	42	49	54	42	0B	C5	D5	F5	E5	43	4C	52	41	20	4F	00	131	
57A0	00	00	43	4C	52	42	20	5F	00	00	00	43	4C	52	00	03	186	
57B0	00	0F	7F	6F	43	4D	50	41	0B	81	91	B1	41	43	4D	50	16D	
57C0	42	0B	C1	D1	F1	E1	43	4D	50	44	47	B3	93	B3	A3	43	1CB	
57D0	4D	50	58	07	8C	9C	BC	43	4D	50	59	47	8C	9C	BC	F0		
57E0	AC	43	4D	50	53	87	8C	9C	BC	43	4D	50	55	87	83	135		
57F0	93	B3	A3	43	4F	4D	41	20	43	00	00	00	43	4F	4D	42	18D	
Sum:	99	02	CE	C2	B5	40	97	1E	76	11	5F	7B	B9	22	C9	D1	1AB	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
5800	20	53	00	00	00	43	4F	4D	00	03	00	03	73	63	43	57	1CB	
5810	41	49	20	3C	00	00	00	44	41	41	00	20	19	00	00	00	1E5	
5820	44	45	43	41	20	4A	00	00	00	44	45	43	42	20	5A	00	1FF	
5830	00	00	00	44	45	43	00	03	00	0A	7A	6A	45	4F	52	41	0BF	
5840	8B	9B	BB	AB	45	4F	52	42	0B	CB	DB	FB	EB	45	58	47	117	
5850	00	10	1E	00	00	00	49	4E	43	41	20	4C	00	00	00	49	1F6	
5860	4E	43	42	20	5C	00	00	00	00	49	4E	43	00	03	00	0C	1B4	
5870	6C	4A	4D	50	00	03	00	0E	7E	6E	4A	53	52	00	03	00	142	
5880	9D	BD	AD	4C	44	41	00	0B	86	96	B6	A6	4C	44	42	00	12D	
5890	0B	C6	D6	F6	E6	4C	44	44	00	07	CC	DC	FC	EC	4C	44	17E	
58A0	58	00	07	8E	9E	BE	AE	4C	44	59	00	47	8E	9E	BE	AE	1BF	
58B0	4C	44	53	00	47	CE	DE	FE	EE	4C	44	55	00	07	CE	DE	15A	
58C0	FE	EE	4C	45	41	5B	01	00	00	30	4C	45	41	59	01	73		
58D0	00	00	00	31	4C	45	41	53	01	00	00	32	4C	45	41	5B		
58E0	55	01	00	00	33	4C	53	4C	41	20	48	00	00	00	4C	169		
58F0	53	4C	42	20	5B	00	00	00	4C	53	4C	00	03	00	0B	7B	1C7	
Sum:	D9	18	77	40	F8	C8	4B	6E	B1	9D	96	F4	AA	7C	05	44	16B	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
5900	68	4C	53	52	41	20	44	00	00	00	00	4C	53	52	42	20	54	1A5
5910	00	00	00	00	4C	53	52	00	03	00	04	74	64	4D	55	4C	00	1BE
5920	20	3D	00	00	00	00	4E	45	47	41	20	40	00	00	00	4E	45	16B
5930	47	42	20	50	00	00	00	4E	45	47	00	03	00	00	00	70	60	1A6
5940	4E	4F	50	00	20	12	00	00	00	00	4F	52	41	00	0B	8A	9A	130
5950	BA	AA	4F	52	42	00	0B	CA	DA	FA	EA	4F	52	43	43	0B	109	
5960	1A	00	00	00	50	53	48	53	10	34	00	00	00	00	50	53	48	187
5970	55	10	36	00	00	00	50	55	4C	53	10	35	00	00	00	50	50	174
5980	55	4C	55	10	37	00	00	00	52	4F	4C	41	20	49	00	00	1D4	
5990	00	52	4F	4C	42	20	59	00	00	00	52	4F	4C	00	03	00	198	
59A0	09	79	69	52	4F	52	41	20	46	00	00	00	52	4F	52	42	1BA	
59B0	20	56	00	00	00	52	4F	52	00	03	00	06	76	66	52	44	1F4	
59C0	49	00	20	3B	00	00	00	52	54	53	00	20	39	00	00	00	1F6	
59D0	53	42	43	41	0B	82	92	B2	A2	53	42	43	42	0B	C2	D2	145	
59E0	F2	E2	53	45	58	00	20	1D	00	00	00	53	54	41	00	03	1EC	
59F0	00	97	B7	A7	53	54	42	00	03	00	D7	F7	E7	53	54	42	181	
Sum:	52	FC	C2	56	C4	BF	09	9D	4D	33	03	C2	DB	D2	07	E2	16A	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
5A00	00	03	00	DD	FD	ED	53	54	58	00	03	00	9F	BF	AF	53	12C	
5A10	54	59	00	43	00	9F	BF	AF	53	54	53	00	43	00	DF	FF	11B	
5A20	EF	53	54	55	00	03	00	DF	FF	EF	53	55	42	41	0B	80	171	
5A30	90	B0	A0	53	55	42	0B	C0	D0	F0	E0	53	55	42	44	A5		
5A40	07	83	93	B3	A3	53	57	49	00	20	3F	00	00	00	53	57	16F	
5A50	49	32	60	3F	00	00	00	53	57	49	33	A0	3F	00	00	00	11F	
5A60	53	59	4E	43	20	13	00	00	00	54	46	52	00	10	1F	00	18B	
5A70	00	00	00	54	53	54	41	20	4D	00	00	54	53	54	42	20	106	
5A80	5D	00	00	00	54	53	54	00	03	00	0D	7D	6D	4F	52	47	13A	
5A90	00	FF	00	00	00	00	00	46	43	42	00	FE	00	00	00	00	10E	
5AA0	43	43	00	FD	00	00	00	00	46	44	42	00	FC	00	00	00	14B	
5AB0	00	45	51	55	00	FB	00	00	00	00	52	4D	42	00	FA	00	1C1	
5AC0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
5AD0	42	43	43	00	24	42	43	53	00	00	25	42	45	51	00	1C1		
5AE0	00	27	42	47	45	00	00	2C	42	47	54	00	00	2E	42	48	1B6	
5AF0	49	00	00															

## 例2 オフセットの例

```

0001 2000          ORG $2000
0002 2000 EC08     LDD B,X
0003 2002 EC887F   LDD 127,X
0004 2005 EC890400 LDD 1024,X
0005 2009          ;
0006 2009 EC890010 LDD $10,X
0007 200D EC8810   LDD <$10,X

```

## 例3 クロス・ファレンス出力例

```

3029 BYTE 0009 0027*
301D DATA 0016 0024*
3021 EXEC 0010 0025*
FC50 FINISH 0013*
3012 LOOP 0018* 0021
ABF4 MONITR 0003* 0037
3023 RETURN 0026*
FD0F ROMRAM 0002* 0033 0036
FC50 TRA 0014* 0017
3029 BYTE 0009 0027*
301D DATA 0016 0024*
3021 EXEC 0010 0025*
FC50 FINISH 0013*
3012 LOOP 0018* 0021
ABF4 MONITR 0003* 0037
3023 RETURN 0026*
FD0F ROMRAM 0002* 0033 0036
FC50 TRA 0014* 0017

```

## RANDOM BOX

## FM-7 独立モニタ用 裏RAM有効利用法

FM-7で“独立モニタ”を裏RAMに入れた場合、ROMに代ってこれられません。無理にROMに切り換えて、リセットしても正常になることは少ないようです。

また、正常にROMに代れても裏RAMにある独立モニタを起動するのに6バイトのプログラムを必要とし、面倒です。そこで作ったのがこのプログラムです。

このプログラムは独立モニタの後ろにつけても位置自由になっているので安心してください。このプログラム中で29行以後はいりません。ただし、独立モニタのアドレス70F0からのデータを16,19,00になおしておいてください。

使い方はROMに代るのに必ずBコマンドを利用します。裏RAM上の独立モニタを起動するには、モニタからGFC50 **[RETURN]**です。独立モニタ上で\$FC50~FC7Fまでを使ってもBコマンドのたびに書くので、有効に利用してください。

このようにした“独立モニタ”と“エディタ・アセンブラ”を利用すると使いやすいですヨ。一度試してみてください。

書き忘れていましたが、独立モニタをどこに置いても起動は\$FC50です。どうですか、便利でしょう。

## 裏RAM上独立モニタ用プログラム

```

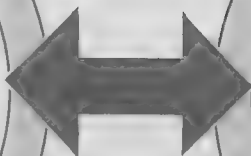
0001 0000          :
0002 0000          : ROMRAM=$FD0F
0003 0000          : MONITR=$ABF4
0004 0000          :
0005 7A00          : ORG $7A00
0006 7A00          :
0007 7A00          :
0008 7A00 1F50     : TFR PC,D
0009 7A02 B37A2A   : SUBD BYTE
0010 7A05 30BD0019 : LEAX EXEC,PC
0011 7A09 EDB4     : STD .X
0012 7A0B          :
0013 7A0B          : TRANCEULATE
0014 7A0B          : TRA= $FC50
0015 7A0B          :
0016 7A0B 30BD000F : LEAX DATA,PC
0017 7A0F 10BEFC50 : LDY TRA
0018 7A13 EC81     : LOOP: LDD .X++
0019 7A15 EDA1     : STD .Y++
0020 7A17 B1FF     : CMPA #$FF
0021 7A19 26F8     : BNE LOOP
0022 7A1B 7EFC56   : JMP $FC56
0023 7A1E          :
0024 7A1E 7FFD0F7E : DATA: FDB $7FFD,$0F7E :FULL RAM
0025 7A22          : EXEC: RMB 0 :EXECUTE ADDR
0026 7A24 B6FD0F7E : RETURN: FDB $B6FD,$0F7E,$ABF4 :ROM MODE & JMP MONITOR
      F4
0027 7A2A 0A02     : BYTE: FDB $A02 :Byte 27
0028 7A2C FF       : FCB $FF
0029 7A2D          :
0030 FC50          : ORG $FC50
0031 FC50          :
0032 FC50          : DUMMY
0033 FC50          :
0034 FC50 7FFD0F   : FINISH:CLR ROMRAM
0035 FC57 7C----   : JMP START
0036 FC56          :
0037 FC56 B6FD0F   : LDA ROMRAM
0038 FC59 7EABF4   : JMP MONITR
0038 FC5C          :

```



# FM-7/8

## FM-8



## FM-7

# マシン語コンバータ

■コンピ

富士通からFM-8の下位バージョンとしてFM-7が発売されました。BASICはコンパチのもののマシン語は互換性がありません。そこでマシン語コンバータを作りました。

## プログラムについて

FM-8のマシン語のプログラムでは、BASICのサブルーチンを使わず、直接サブCPUをコントロールして画面をかかせているものがほとんどで、I/O誌の“INVADER”ゲームなどは、そのままFM-7で走ります。

しかし、1文字出力など簡単なものは、BASIC ROMのサブルーチンが使っていたりするので、BASIC ROMに飛ばすサブルーチンを変えれば、FM-7でも、ほとんどが使用可能です。したがって、比較的使われそうなものをコンバートしようと、このプログラムを作りました。

プログラム自体は\$6000～\$681F (EXEC & H6000) です。また、実行中にBコマンドでBASICへ戻ると暴走するので、STOP+RESETでBASICに戻ってください。

コンバート表

FM-8	FM-7	内 容
947C	8FD0	END エントリ
948D	8FE1	BREAK エントリ
9F97	9B57	改行する。
9FDF	9BDB	Xレジスタのアドレスから0があるまで文字を出力する。
A01A	9C16	XレジスタのアドレスからBレジスタの回数だけ文字を出力する。
AFE3	AC37	アキュムレータDの内容を16進数で出力。
AFE9	AC3D	アキュムレータAの内容を16進数で出力。
B96D	B615	アキュムレータDの内容を10進数で出力。
D336	D072	キーが押されるまで待ち、押された値をAレジスタに入れる。
D352	D08E	Aレジスタのキャラクタを出力。
D9B8	D76F	MOTOR ON
D9BB	D772	MOTOR OFF
DA50	D807	一行入力、\$043D番地からASCIIコードで入力、ストップ・キーが押された

DD69	DB6D	ときはCフラグをセットする。
F2D8	F17D	キーが押されていたら、その値をAレジスタに入れる。
AFA2	ABF4	押されていないときはZフラグをセットする。
D8B8	8E63	BIOS エントリ モニタ・エントリ BASICホット・スタート

以上のものとコンバートします。

まず、プログラムを走らせると、現在使っているマシンがFM-8かFM-7か判断して、サブルーチン・アドレスをそのマシン語アドレスに変更するので、本プログラムはFM-8、7どちらでも走ります。

具体的な使い方は、スタート直後、“HIT ANY KEY”と表示するので、何かキーを押し、続いてアドレスを聞いてくるので16進数で入力してください。

\$7E (ジャンプ)、\$BD (サブルーチン・ジャンプ)を見つけると、予約アドレスかどうか照合して、もしそうなら“CHANGE ADDRESS ~”と表示し、違うときは、それがBASIC ROMのアドレスかどうか調べて、そうなら“ADDRESS~”と表示します。

なお、\$7E、\$BDがデータかプログラムかは区別していないので、データを書き換える可能性もあります。

予約アドレスを追加したい場合は、\$6980～\$69BFがFM-8、\$69C0～\$69FFがFM-7のアドレス・テーブルになっているので、そこに追加してください。

そのとき最後の0を残しておいてください (14個追加できます)。

注) FM-8からFM-7への変更はそのままで走りますが、FM-7からFM-8のときは、パレット機能、サウンド機能などの関係で、すべて走るというわけではありません。

なお、プログラムは完全ポジション・インディペンデントです。

## FM-7を使ってみて

サブCPUのVRAMアクセス・フラグ、CRTのON/OFFのアドレスはFM-8と同じです。また、直接サブCPUをHALTしてコマンドを書きこんでも、まったく同じに使える。

ます、つまり、FM-7はBASIC ROMのサブルーチンさえ使わなければ、FM-8とマシン語コンパチです。

また、BASIC ROMもほとんどFM-8と同じようです(アドレスは違いますが)。

私たちFM-8のユーザーにとっては、パレット機能やサウンド機能、マルチ・ページなど、とてもくやしい気がします。

### プログラム・ダンプ・リスト

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6000 B6 84 00 27 31 31 8D 07 77 10 AF 8D 07 4D 31 8D :2E
6010 09 AE 10 AF 8D 07 59 86 38 A7 8D 01 8A 86 37 A7 :44
6020 8D 01 9D A7 8D 04 9F 86 39 A7 8D 04 7E CC 3D 3E :8E
6030 30 8D 00 AF 20 2F 31 8D 09 86 10 AF 8D 07 1C 31 :A8
6040 8D 09 3D 10 AF 8D 07 28 86 37 A7 8D 01 59 86 38 :57
6050 A7 8D 01 6C A7 8D 04 6E 86 35 A7 8D 04 4D 30 8D :44
6060 00 77 CC 3C 3D ED 8D 01 9A ED 8D 01 AA ED 8D 01 :71
6070 C2 ED 8D 01 DA ED 8D 02 ED 8D 02 2B ED 8D 02 :B8
6080 4E 8D 02 64 ED 8D 02 83 ED 8D 02 A3 ED 8D 02 :C0
6090 D9 ED 8D 03 1C ED 8D 03 31 ED 8D 03 47 ED 8D 03 :61
60A0 76 EC 81 ED 8D 04 9C ED 8D 06 48 ED 8D 06 94 EC :C5
60B0 81 ED 8D 05 CC ED 8D 05 F8 EC 81 ED 8D 04 2A ED :45
60C0 8D 04 25 EC 81 ED 8D 04 75 EC 8D 00 5D ED :4C
60D0 8D 04 26 ED 8D 05 23 2D 24 84 8D AF E3 D3 36 DA :33
60E0 50 F2 D8 8F E1 AC 37 D0 72 D8 07 F1 7D B6 FD 05 :B4
60F0 2B F8 B6 80 B7 FD 05 B6 FD 05 2A F8 39 8D EE 30 :A6

Sum: 1D 64 15 C4 57 C5 0A DC DD 53 66 C5 A0 20 E7 45 :A3

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6100 8D 00 12 10 8E FC 82 C6 09 A6 80 A7 A0 5A 26 F9 :70
6110 7F 0D 05 20 09 01 00 50 19 00 01 00 00 30 8D :E4
6120 00 0F 31 8D 00 13 10 AF 8D 00 08 BD F2 D8 16 03 :D4
6130 B7 14 14 61 39 03 AD 00 00 46 55 4A 49 54 53 55 :53
6140 20 4D 49 43 52 4F 20 3B 20 3D 3C 3E 20 46 :6F
6150 55 4A 49 54 53 55 20 4D 49 43 52 4F 20 37 0A :ED
6160 09 4D 41 43 48 49 4E 45 20 4C 41 4E 47 55 41 :47
6170 45 20 43 4F 4E 56 45 52 54 55 52 50 56 65 72 :2E
6180 20 31 2E 32 20 27 38 33 2E 32 2E 31 20 62 79 :3D
6190 20 45 49 5A 49 0A 0D 0A BA C9 20 CC DF DB 88 DE :31
61A0 D7 D1 20 CA 46 4D 20 3B 20 C9 20 CF BC DD BA DE :93
61B0 20 C9 20 CC DF DB 88 DE D7 D1 20 A6 20 46 4D 2D :73
61C0 37 20 D6 83 C6 20 CD DD B6 DD 20 BD D9 20 CC DF :84
61D0 DB 88 DE D7 D1 20 C3 DE BD A1 0D 03 C2 B7 DE C9 :F4
61E0 D3 C9 20 C4 20 BF C9 20 C9 20 0A C9 C5 C4 DE :6F
61F0 A6 20 CD DD B6 DD 20 BC CF BD A1 0D 0A 24 39 34 :B4

Sum: 48 F5 CA 94 06 B4 A4 70 D2 75 97 BE 31 75 64 C2 :A1

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6200 37 43 20 3E 3E 20 24 38 46 44 30 09 45 4E 44 :0A
6210 0D 24 39 34 38 44 20 3D 3E 20 24 38 46 45 31 :09
6220 42 52 45 41 4B 20 65 6E 74 42 79 0A 0D 24 39 :46
6230 39 37 20 3D 3E 20 24 39 42 75 37 09 70 72 69 :6E
6240 74 20 43 52 2C 43 46 0A 0D 24 39 46 44 46 20 :3D
6250 3E 20 24 39 42 44 42 09 58 72 65 67 20 B6 D7 :20
6260 30 20 B6 DE 20 B1 D9 CF C3 DE 20 D3 BC DE 20 :A6
6270 20 72 69 6E 74 0A 0D 24 41 30 31 41 20 3D 3E :06
6280 20 24 39 43 31 36 09 58 72 65 67 20 B6 D7 20 :42
6290 72 65 67 20 C9 20 B6 B2 BD 83 20 70 72 69 6E :74
62A0 0A 0D 24 41 46 45 33 20 3D 3E 20 24 41 43 33 :07
62B0 09 70 72 69 6E 74 20 44 20 69 6E 20 68 65 78 :0A
62C0 0D 24 41 46 45 39 20 3D 3E 20 24 41 43 33 44 :09
62D0 70 72 69 6E 74 20 41 72 65 67 20 69 6E 20 68 :65
62E0 78 0A 0D 24 42 39 36 44 20 3D 3E 20 24 42 36 :31
62F0 35 09 70 72 69 6E 74 20 44 20 69 6E 20 64 65 :63

Sum: 90 6F AA 18 0D 5F 55 8C 19 63 F2 11 2F 04 EB 01 :AC

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6300 69 6D 61 6C 0A 0D 24 44 33 35 32 20 3D 3E 20 :24
6310 44 30 38 45 09 41 72 65 67 20 C9 20 B7 AC D7 :B8
6320 C0 20 A6 70 72 69 6E 74 0A 0D 24 44 33 33 36 :20
6330 3D 3E 20 24 44 30 37 32 09 85 BB DA C0 20 B7 :2D
6340 20 C9 20 C5 B2 B6 83 20 A6 20 41 72 65 67 20 :C6
6350 20 B2 DA D9 20 20 2B 85 BB DA D9 CF C3 DE 20 :CF
6360 C2 29 0A 0D 24 44 36 39 20 3D 3E 20 24 44 42 :82
6370 36 44 09 B5 BB DA C0 20 B7 2D 20 C9 20 C5 B2 :D6
6380 B3 20 A6 20 41 72 65 67 20 C6 20 B2 DA D9 20 :E7
6390 2B 85 BB DA C3 B2 C5 B2 20 C4 B7 CA 20 5A 66 :6C
63A0 61 67 20 A6 20 BE AF CA 29 0A 0D 24 44 39 42 :38
63B0 20 3D 3E 20 24 44 37 36 49 42 4D 4F 54 4F 52 :20
63C0 4F 4E 0A 0D 24 44 39 42 42 20 3D 3E 20 24 44 :37
63D0 37 32 09 4D 4F 54 4F 52 40 4F 46 46 0A 0D 24 :44
63E0 41 35 30 20 3D 3E 20 24 44 38 30 37 09 31 B7 :DE
63F0 AE B3 20 C6 AD B3 DB AE B8 20 20 2B 24 30 34 33 :08

Sum: B3 C4 8E A5 1F AA AA F3 0B C2 55 78 3B B8 B7 46 :67

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6400 44 20 B6 D7 20 41 53 43 49 BA 2D C4 DE 29 0A :36
6410 0D 24 46 32 44 38 20 3D 3E 20 24 46 31 37 44 :09
6420 42 49 4F 53 20 65 6E 74 72 79 0A 0D 0A CF BC :D0
6430 BA DE 09 C9 20 CF DF DB 88 DE D7 D1 20 0A :46
6440 C2 B6 CD DA D9 20 42 41 53 49 43 20 53 75 62 :20
6450 72 6F 75 74 69 6E 65 20 C9 20 B5 D3 C5 20 C9 :18
6460 CA 20 B3 B4 C9 20 D3 C9 C3 DE BD A1 0A 0D BC :C0

```

ます。

最後にFM-7を気おねなく使わせてくれた東海アビックさん、どうもありがとうございました(ちなみに、この店はアフターサービスが、信じられないほどいいです)。

### 参考文献

- 1) "FM-8 活用研究", I/O別冊

```

6470 B6 DE AF C3 A4 CE C4 DD C4 DE 20 C9 20 CC DF DB :4A
6480 B8 DE D7 D1 20 CA 20 BA DD CA DE 20 C4 20 C3 DE :39
6490 B7 CF BD B6 DE A4 C5 B6 C6 CA 20 C3 DE B7 C5 B2 :75
64A0 D3 C9 D3 B1 DB CF BD A1 0A 0D 28 39 30 25 20 B2 :C4
64B0 BC DE AE 83 20 CA 20 BA DD CA DE 2D C4 20 BD DA :EC
64C0 CA DE 20 46 4D 2D 37 20 C3 DE 20 C2 B6 B4 CF BD :58
64D0 A1 29 0A 0D 09 09 48 49 5A B6 04 3D 27 55 B6 :73
64E0 45 59 20 21 21 00 00 00 BD D3 36 30 8D 00 0F :C3
64F0 BD 00 13 10 AF 8D 00 08 BD F2 D8 16 00 D7 14 14 :90

Sum: 3C 42 90 59 6F F0 00 11 64 47 E6 4D 88 15 4E 0A :AA

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6500 65 06 00 2A 00 00 0A 0D 0A 53 54 41 52 54 20 :A5
6510 44 44 52 45 53 53 20 A6 20 B2 DA C3 20 B8 C0 DE :70
6520 B8 B2 20 28 31 36 BC DD 20 34 B9 C0 29 3F 20 :2A
6530 00 CC 00 00 FD 04 3D FD 04 3F FD 04 41 BD DA 50 :73
6540 24 03 7E 94 BD B6 04 41 26 5A B6 04 3D 27 55 B6 :73
6550 04 3D 80 30 C6 10 3D E7 8D 00 47 B6 04 3E 80 :30
6560 B1 09 23 02 80 07 B1 0F 2E 3A AB 8D 00 3A A7 :8D
6570 00 30 B6 04 3F 80 30 B1 09 23 02 80 07 B1 0F :2E
6580 23 C6 10 3D E7 8D 00 1B B6 04 40 80 30 B1 09 :23
6590 02 80 07 B1 0F 10 2E 00 AB 8D 00 06 A7 8D 00 :D4
65A0 02 39 00 00 17 FB 46 30 8D 00 18 10 8E FC B2 :C6
65B0 11 A6 80 A7 0A 5A 26 F9 7F FD 05 86 01 A7 8D :00
65C0 01 39 00 03 0F 0A 11 02 45 52 52 4F 52 20 21 :55
65D0 0A 0D 11 07 00 6F 8D FF ED 8D 17 FF 55 A6 8D :F2
65E0 10 26 FF 07 EC BD FF BA ED 8D 01 3A 30 8D 00 :E2
65F0 31 8D 00 12 10 AF 8D 00 07 BD F2 D8 20 31 14 :23

Sum: 91 5F F0 E9 4B B1 D9 44 27 BE BC 5B 31 39 5D 3E :83

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6600 66 06 00 28 00 00 0A 45 4E 44 20 20 41 44 44 :9E
6610 52 45 53 53 20 A6 20 B2 DA C3 20 B8 C0 DE B8 :55
6620 20 28 31 36 BC DD 20 34 B9 C0 29 3F 20 00 6F :2C
6630 8D FF 8F 17 FE FB A6 BD FF B8 10 26 FE AD EC :8D
6640 FF 6D ED 8D 00 E2 10 A3 8D 00 DB 2E 06 17 FF :54
6650 16 FE 98 17 FA 97 30 8D 00 12 10 8E FC B2 C6 :0F
6660 A6 80 A7 0A 5A 26 F9 7F FD 05 20 10 03 0D 0A :8B
6670 53 54 41 52 54 20 A5 A5 A5 20 24 0C EC 8D 00 :A6
6680 BD AF E3 17 FA 67 30 8D 00 12 10 8E FC B2 C6 :10
6690 A6 80 A7 0A 5A 26 F9 7F FD 05 20 10 03 0D 0A :88
66A0 20 20 45 4E 44 20 A5 A5 A5 20 24 0C EC 8D 00 :78
66B0 BD AF E3 17 FA 37 30 8D 00 12 10 8E FC B2 C6 :1A
66C0 A6 80 A7 0A 5A 26 F9 7F FD 05 20 18 03 18 0A :D4
66D0 0A D6 DB 8C B2 20 C3 DE BD B6 20 2B 29 00 6F :72
66E0 20 4E 29 20 3F 20 0D BD D3 36 81 59 27 07 B1 :4E
66F0 26 F5 7E 94 BD 17 F9 F5 30 8D 00 12 10 8E FC :8A

Sum: A9 3B 5B BA EC 9E 81 59 6E 4D CD C4 B8 BA 5C 16 :FD

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6700 C6 17 A6 80 A7 0A 5A 26 F9 7F FD 05 20 1E 03 :9A
6710 0A 0D 0A 0A 0A 52 75 6E 6E 69 6E 67 2E 2E 2E :C5
6720 0A 0D 0A 0A 0A 00 00 00 00 00 00 00 AE BD FF :6E
6730 A6 84 B1 7E 27 0F B1 BD 27 08 A6 80 AC BD FF :15
6740 23 EE 7E 94 BD A7 8D 00 F8 A6 80 AF 8D FF D7 :EC
6750 84 ED 8D FF D5 ED 8D 00 E9 10 8E 00 00 4F AE :76
6760 10 27 01 53 AC BD FF C2 27 04 4C 4C 20 F0 10 :F6
6770 00 00 AE A6 1F 10 AE BD FF AC AF 8D 00 C1 ED :81
6780 AF 8D FF A2 ED 8D 00 BC A6 8D 00 B5 B1 7E 27 :09
6790 30 8D 00 B5 C6 07 20 06 30 8D 00 AA C6 06 E7 :8D
67A0 00 BD C6 03 31 8D 00 EA A6 80 A7 AB 13 A7 0A :65
67B0 26 F6 A6 8D 00 8C BD 74 A7 8D 00 DB A6 8D 00 :82
67C0 BD 72 A7 8D 00 D2 A6 8D 00 79 8D 60 A7 8D 00 :98
67D0 A6 8D 00 6F BD 5E A7 8D 00 C0 A6 8D 00 66 BD :4C
67E0 A7 8D 00 C6 A6 8D 00 5C BD 4A A7 8D 00 BD A6 :84
67F0 00 53 BD 38 A7 8D 00 B4 A6 8D 00 49 BD 36 A7 :73

Sum: 16 71 94 7F CD 29 11 EA EB 90 9B 19 B9 03 39 62 :E1

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6800 00 AB A6 8D 00 39 BD A7 8D 00 76 A6 8D 00 2F :D4
6810 BD 22 A7 8D 00 6D A6 8D 00 26 BD 10 A7 8D 00 :64
6820 A6 8D 00 1C BD 0E A7 8D 00 58 20 20 44 44 44 :C9
6830 B8 30 20 04 B4 0F B8 30 B1 39 23 02 88 07 39 :00
6840 00 00 00 00 00 00 4A 4D 50 4A 53 52 17 F8 9E :30
6850 BD 00 17 10 8E FC B2 C6 A4 A6 80 A7 0A 5A :B6
6860 7F FD 05 AE 8D FE BF 16 FE D2 03 48 11 00 43 :48
6870 41 4E 47 45 20 41 44 44 52 45 53 53 20 A5 A5 :50
6880 20 24 30 30 30 20 20 20 20 20 43 4F 4D 4D 41 :3F
6890 44 20 4A 4D 50 20 24 58 58 58 58 20 20 3D 3D :C9
68A0 3D 3E 20 20 20 4A 4D 50 20 24 58 58 58 58 0A :7D
68B0 11 07 00 00 00 00 00 AE 8D FE 6B EC 84 10 B3 :7F
68C0 FF 10 23 FE 67 A6 1F 81 7E 27 06 30 BD FF 7A :20
68D0 04 30 BD FF 71 31 BD FF B9 C6 03 A6 80 A7 0A :3E
68E0 26 F9 A6 BD FF 5C 17 FF 43 A7 BD FF AA A6 BD :15

```





FM-7/8

# サウンド・エディタ

## ver.2.0



現在COMPACではFM-8用PSG基板と、そのサポート・ソフトとしてサウンド・エディタを発売していますがFM-7ではこのPSGが標準装備になりました(FM-8のPSG基板はFM-7とフルコンパチブルです)。そこでサウンドもFM-7に対応できるように変更し、バージョン2.0として発表します。

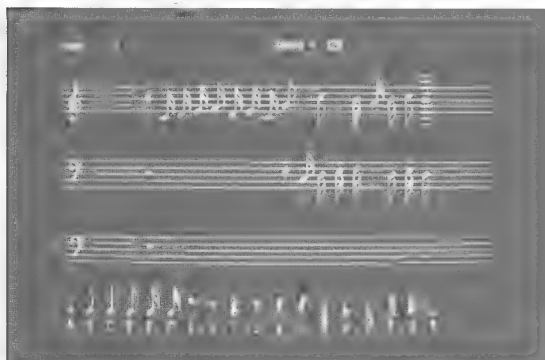
### 特 徴

このプログラムは、PSGのミュージック・データ作成用に作られたもので、音楽の自動演奏などに適しています。工夫しだいではゲームの効果音サブルーチンの作成などにも利用できます。特徴を次にあげます。

- ①楽譜による編集を行ない、それを演奏コードと呼ばれるマシン語プログラムに変換する。
- ②変換された演奏コードは、完全にリロケータブルな独立したプログラムで、他のプログラムのサブルーチンとしても利用できる(COMPACから発売されているGALAXY FLYも、このルーチンを一部使っています)。
- ③インターバル・タイマによる割り込みを採用しているため、他のプログラムと同時実行ができる。つまりメインの側のプログラムで、音楽演奏を意識せずにプログラミングすることができる。

このようにミュージック・サブルーチン作成のためのツールとして利用する他に、入力した譜面を演奏させて、単に画面上に表示されるデモを眺めるだけでも、十分に楽しむことができます。

写真1 エディタ・コマンド・モード



### プログラムの入力

カセットを利用している人は、リスト1(BASIC)、リスト2(マシン語)、リスト3(マシン語)を入力して、それぞれ“SYS”、“SEQ”、“SUB”というファイル名を付け、1本のテープに順にセーブしておきます。プログラムを起動させるには、先頭のBASICプログラム“SYS”をロードRUNさせれば、自動的に“SEQ”、“SUB”を読み込んで実行が開始されます。

ディスクを使っている人は、そのままではメモリ容量が足りないので少々やっかいです。そこでプログラムのうち必要な部分だけを、その都度ディスクからロードして実行させることにします。実際には次のようにしてください。

- ①リスト1に次の部分を追加、変更して“SYS”というファイル名でディスクにセーブする。

追加部分：14850 LOADM “EAT”  
15150 LOADM “GEN”  
15450 LOADM “DSP”

変更部分：10700行

GENADR=&H7004→GENADR=&H5000  
DSPADR=&H4004→DSPADR=&H5000

- ②リスト2を入力し、“SEQ”というファイル名でディスクにセーブする。

写真2 演奏中



③リスト3のうち\$4500～\$4FFFの部分を入力し、“SU B”というファイル名でディスクにセーブする。

④リスト3のうち\$5000～\$6FFFまでを入力し、“ED7”というファイル名でディスクにセーブする。

⑤リスト3の\$4000～\$44FFまでを\$5000～\$54FFへリロケートして入力し、“DSP”という名でセーブする。

⑥リスト3の\$7000～\$70FFまでを\$5000～\$50FFへリロケートして入力し、“GEN”という名でディスクへセーブする。

これで、ディスクでもサウンド・エディタが使えるようになります。プログラムの起動はカセット版と同じく“SYS”をロード、RUNすることで行ないます。

## 使用法

### ①システム・コマンド・モード

プログラムを起動すると、システム・コマンド・モード(コマンド・ラインの左端に“SYS”と表示される)になります。ファイルの入出力や、エディタおよびジェネレータの起動は、このシステム・コマンド・モードから行なうことになります。各コマンドの先頭の英大文字1字をキーインすることでコマンドを指定してください。

#### ●E (dit)

エディタを起動します。楽譜データの入力や編集はこのエディタで行ないます(写真1.エディタ・コマンド・モードの欄を参照)。

#### ●G (en)

ジェネレータを起動します。楽譜データを演奏コードに変換するのに、このジェネレータを使います(ジェネレータ・コマンド・モードの欄を参照)。

#### ●P (lay)

ジェネレータで生成された演奏コードを実際に演奏します。このコマンドを実行後、画面上にピアノの鍵盤が表示され、演奏と同時に各パートの出ている音の高さをディスプレイします(写真2)。P (lay コマンドを使う場合は前もってジェネレータで演奏コードを生成しておくが、演奏コードをファイルからメモリ上に読み込んでおかなければなりません。

#### ●D (ir)

ファイルのディレクトリを取ります。このコマンドを実行すると、What File ?と聞いてくるので、ディレクトリを取ろうとするファイル・ディスクリフタを入力します。

D (ir コマンドは、ディスクやバブルを使っている場合にのみ有効です。もし誤ってテープ・ファイルに対してこのコマンドを入力してしまった場合は、一度[BREAK]キーを押してから、RUNでプログラムを再起動してください。この場合でもメモリ上にある楽譜データや演奏コードは消えることはありません。

#### ●L (oad)

楽譜データをファイルからロードします。L (oad コマンドを実行すると、“Through away current buffer ?”と確認をとってきます。これは、『現在メモリ上にある楽譜データをクリアしてもいいか?』ということなので、新たに楽譜データをロードするのであれば[Y]とキーインします。すると、What File ?とファイル名を聞いてきます。ここでファイル・ディスクリフタを入力すれば、楽譜データがメモリ上にロードされます。

#### ●S (ave)

楽譜データをファイルにセーブします。S (ave コマンドを実行すると、What File ?とファイル名を聞いてくるので、ファイル・ディスクリフタを入力してください。

#### ●R (ead)

演奏コードをファイルからロードします。L (oad, S (ave と同様にファイル名を聞いてくるので、ファイル・ディスクリフタを入力してください。

#### ●W (rite)

ジェネレータで生成した演奏コードをファイルにセーブします。ファイル名を要求してきたら、ファイル・ディスクリフタを入力します。

#### ●K (ill)

ディスクやバブル上にあるファイルを消去します。“What File ?”に対してファイル・ディスクリフタを入力すると、“Kill [ファイル・ディスクリフタ] ?”という確認をとってくるので、[Y]またはその他のキーを入力してください。[Y]を入力した場合にのみファイルが消去されます。

またこのコマンドはD (ir コマンドと同様に、テープ・ファイルについては使えません。

#### ●N (ew)

メモリ上の楽譜データをクリアします。“Through away current buffer ?”という確認をとってくるので、本当にクリアしていいのなら[Y]を、そうでなければそれ以外のキーを押してください。

システム・コマンド・モードでエラーが発生した場合はそのエラー番号を画面上に表示します。このエラー番号はBASICのエラー番号と同じなので、エラーの内容を知りたい場合は、BASICの文法書を参照してください。

### ②エディタ・コマンド・モード

楽譜データの編集を行なう場合に、このエディタを使います。システム・コマンド・モードのE (dit コマンドでエディタが起動し、エディタ・コマンド・モードになります。エディタでは次のような機能が使用可能です。

①ページ単位の編集(挿入、削除、変更)。ここでいうページとは原則として、楽譜上の1小節に相当するもので、各ページに先頭から通し番号を付けて処理します。つまり第nページといった場合は、先頭からn小節を意味することになります。

②ハート単位の編集(挿入、削除、変更)。五線譜上の音符、記号などをカーソルを使って編集することができます。パートは3つあり、画面上に表示される譜面の上から順番にパート1、パート2、パート3という名前が付いています。

このエディタで編集する楽譜データは、新しいデータを作成する場合をのぞき、システム・コマンド・モードのL (oadコマンドを使って、エディタの起動前にメモリ上に読み込んでおかなければなりません。

以下にエディタ・コマンド・モードで使用できるコマンドを解説します。コマンドの指定はシステム・コマンドと同様に、先頭の英大文字1字で指定しますが、一部でパラメータを必要とするコマンドがあるので、それらについてはコマンドの解説の部分で特にお示ししておきました。

#### ●E (dit 指定方法: nE (1 ≤ n ≤ 3)

各パートを編集するときに使います。エディタ・コマンド・モードで、パートに対応する数字を入力し、それに続けて[E]を押すと、そのパートにカーソルが表示されます。

つまり**I**と入力した場合は、パート1の部分にカーソルが表示され、そのパートの編集ができるようになります。

音符および記号の入力は、**↑****↓**キーで音の高さを合わせて**A**～**B**のキーを押します。すると画面下の方に表示されている記号のうちから、それらのキーに対応する記号がカーソルの位置に挿入されます。一度入力した記号を他の記号に置き換える場合は、**←****→**キーでその位置にカーソルを合わせ、一度**DEL**キーを押してその記号を削除してから、また新たに入力し直します。その他各キーの機能などについては、表1を参考にしてください。

こうしたパートの入力が1ページ分終わったら**ESC**キーでエディタ・コマンド・モードに戻ります。

#### ●N (ext)

現在画面上に表示されているページの次のページを表示します。

#### ●P (re)

現在画面上に表示されているページの前のページを表示します。

#### ●J (ump 指定方法: nJ (1 ≤ n))

任意のページを画面に表示します。たとえば3ページめを表示させたいのなら**3J**と入力してください。

#### ●I (nsert)

現在表示しているページの前に1ページ分挿入します。結果として、新たに挿入したページより後方のページは、そのページの番号が変化します。

#### ●D (elete)

現在表示しているページを削除します。結果として削除したページより後方のページは、そのページ番号が変化します。

#### ●V (iolin 指定方法: nV (0 ≤ n ≤ 3))

任意のパートにト音記号( $\text{♩}$ )をセットします。**3V**と入力したならば、パート3にト音記号がセットされます。なおパートに0を指定すると、パート1、2にト音記号、パート3にヘ音記号をセットします。

#### ●B (ass 指定方法: nB (0 ≤ n ≤ 3))

任意のパートにヘ音記号( $\text{♮}$ )をセットします。セットの仕方は**V** (iolinコマンド)と同じで、 $n$ に0を指定した場合も同様です。

#### ●F (lat 指定方法: nF (0 ≤ n ≤ 7))

変種音階に調号をセットします。譜面の左端にあるフラットの数が $n$ です。仮に変ロ長調にセットしたければ**1F**と入力します。

#### ●S (harp 指定方法: nS (0 ≤ n ≤ 7))

嬰種音階に調号をセットします。指定方法は**F**(latコマンド)と同じです。

#### ●T (empo 指定方法: nT (1 ≤ n ≤ 255))

テンポを $n$ にセットします。テンポは演奏中いつでも変更することができます。

#### ●R (peat 指定方法: nR)

繰り返しの途中で指定された回数分のときだけそのページを演奏します。これは記号の「1.」、**2.**などに相当するものです。セットの方法は、たとえば「1.2.3.4」にセットしたい場合、**1234R**と入力します。

サウンド・エディタでは繰り返し記号(「 $\text{♩}$ 」)を指定すると、無限にその部分を繰り返してしまいます。それを避けるために、最後のページに「 $\text{♩}$ 」をセットしてください。

#### ●> (右向き不等号)

繰り返し記号「 $\text{♩}$ 」をセットします。もう一度このコマン

ドを入力すると、この記号はリセットされます。

表1 カーソル・エディット・モード

(エディタ・コマンド・モードからnEコマンドで起動する)

キー	機 能
<b>A</b> ～ <b>R</b>	画面上に表示された記号をカーソルの位置に挿入する
<b>DEL</b>	カーソルの位置の記号を削除する
<b>←</b> <b>→</b>	カーソルの左右方向の移動
<b>↑</b> <b>↓</b>	カーソルの上下方向の移動 (音の高さの決定)
<b>SPACE</b>	カーソルの位置に空白を挿入
<b>ESC</b>	エディタ・コマンド・モードに戻る

\***P**および**Q**キーに対応する記号は、それ以降の音程を臨時的に1オクターブ上げる (あるいは下げる) ことを意味する。ただし、これらの記号を入れ囲にすることはできない。

\***R**キーの記号はタイで、次に続く音がその前の音につながっていることを意味する。

\*付点音符の入力は**B**から**R**までの音符キーと同時に**SHIFT**キーを押すことにより行なう。ただし付点休符の入力はできないので、2つの休符を使って代用する。

\*3連符の入力は、付点音符のときと同様に**GRAPH**キーを押しながら行なう。

表2 システム・コマンド一覧

コマンド	機 能
<b>E</b>	楽譜データの作成、編集 (エディタの起動)
<b>G</b>	楽譜データを演奏コードに変換 (ジェネレータの起動)
<b>P</b>	演奏コードを実行 (演奏) すると同時に、CRT上にデモを表示
<b>L</b>	楽譜データをファイル (テープやディスクなど) からロード
<b>S</b>	楽譜データをファイルにセーブ
<b>R</b>	演奏コードをファイルからロード
<b>W</b>	演奏コードをファイルにセーブ
<b>N</b>	メモリ上の楽譜データをクリア
* <b>D</b>	ファイルのディレクトリをとる
* <b>K</b>	ファイルを消去
*については、テープは使用できません	

表3 エディタ・コマンド一覧

コマンド	機 能
<b>nE</b>	第 $n$ パートを編集 ( $n$ は数字で1 ≤ $n$ ≤ 3)
<b>N</b>	次のページを表示
<b>P</b>	前のページを表示
<b>nJ</b>	$n$ ページめを表示
<b>I</b>	現在表示されているページの前に1ページを挿入
<b>Q</b>	現在表示されているページを削除
<b>nV</b>	第 $n$ パートにト音記号( $\text{♩}$ )をセット、 $n$ に0を指定すると、すべてのパートの音部記号はデフォルトにリセット
<b>nB</b>	第 $n$ パートにヘ音記号( $\text{♮}$ )をセット、 $n$ に0を指定した場合は上記の場合と同じ
<b>nS</b>	調号をセット、 $n$ はシャープ( $\sharp$ )の数、0でリセット
<b>nF</b>	調号をセット、 $n$ はフラットの数、0でリセット
<b>&gt;</b>	繰り返し記号(「 $\text{♩}$ 」)をセット、もう一度同じコマンドを入力するとリセット
<b>&lt;</b>	繰り返し記号(「 $\text{♩}$ 」)をセット、リセット
<b>.</b>	終止線(「 $\text{  }$ 」)をセット、リセット
<b>nT</b>	テンポを $n$ にセット
<b>nR</b>	繰り返しの中で指定された回数分のときだけそのページを演奏「1. 2. 3.」に相当する。 $n=123$ であれば、「1.2.3.」にセット、 $n=0$ ならリセット
<b>Q</b>	システム・コマンド・モードに戻る

表4 ジェネレータ・コマンド一覧

コマンド	機 能
<b>G</b>	楽譜コードを演奏コードに変換
<b>T</b>	トーン・コントロール・スイッチをセット
<b>E</b>	エンベロープ・コントロール・スイッチ、波形、周期のセット
<b>N</b>	ノイズ・コントロール・スイッチ・周波数のセット
<b>L</b>	各チャンネルの音量をセット
<b>Q</b>	システム・コマンド・モードに戻る

#### ●< (左向き不等号)

繰り返し記号「 $\text{♩}$ 」をセットします。もう一度押すとリセッ



トされます。

### ●. (ピリオド)

終止線IIをセットします。もう一度押すとリセットされます。

### ●Q (uit)

エディタ・コマンド・モードを抜け、システム・コマンド・モードに戻ります。

エディタで編集できる楽譜の大きさは、32～48小節ぐらいですが、1ページに2小節を詰め込めば、多少大きなデータを編集することができます。オーバーフローしてしまった場合は、編集機能が動かなくなるので、注意してください。

それからデータが大きいと、ジェネレータを使ったときに楽譜データを破壊してしまう可能性があるのです。編集した楽譜データは、ジェネレートする前にファイルにセーブするよう心がけてください。

## ③ジェネレータ・コマンド・モード

エディタで編集した楽譜データは、そのままでは演奏することができません。そこでその楽譜データを実行可能なマシン語プログラムに変換するのがこのジェネレータです。

ジェネレータでは、ジェネレートする前にパラメータをセットすることで、PSGのセッティングをすることができます。

### ●T (one)

PSGのトーン・コントロール・スイッチをセットします。通常は各パートともenable (ON) にセットされています。

T (one)コマンドを実行すると、次のように聞いてくるので、そのパートのトーン・コントロール・スイッチをONにしたいならば[E](enable)を、OFFにしたいならば[D](disable)を入力します。

PSG-A enable=D (トーン・スイッチをOFFにする)  
PSG-B enable=E (トーン・スイッチはONのまま)  
PSG-B enable=E

### ●N (oise)

PSGのノイズ・コントロール・スイッチとノイズ周波数をセットします。スイッチのセットの仕方はT (one)コマンドと同じで、通常どのパートもdisable (OFF) にセットされています。

PSG-A disable=E (ノイズ・スイッチをONにする)  
PSG-B disable=D  
PSG-C disable=D

Noise Frequency (×100Hz) ? 50

(ノイズの平均周波数を5 kHzにセットする)

ノイズの平均周波数は実際の周波数に1/100をかけて入力してください。入力できるのは400Hzから76,800Hzまでです。

### ●L (evel)

各パートの音量をセットします。音量は0から15までの数字で指定できます。数字が0に近づくほど小さくなり、逆に15に近づくほど大きくなります。通常この値は12にセットされています。

PSG-A 12=15 (指定する音量を入力)  
PSG-B 12=15  
PSG-C 12=15

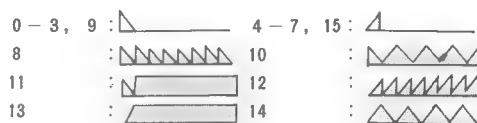
### ●E (nvelope)

PSGのエンベロープ・スイッチと波形、エンベロープ・タイムをセットします。スイッチのセットは他の場合と同様で、通常はOFFになっています。

PSG-A disable=E (パート1にエンベロープをかける)

PSG-B disable=D (その他のパートはそのまま)

PSG-C disable=D



Wave Form ? 0 (波形0を選択)

Envelope Time (X0.01s) ? 100

(エンベロープ・タイムを1秒にセット)

エンベロープ・タイムは、実際の長さに100をかけて入力します。入力できる範囲は0.01sから6sまでです。またエンベロープ・スイッチをONにした場合、そのパートの音量指定は無視されます。

ところでPSGにはエンベロープ・ジェネレータが1つしかありません。そこですべてのパートにエンベロープをかけると、音が出されるたびにトリガがかかることになり、音がとても聞きづらくなる場合があります。この場合はエンベロープをかけるのは、主旋律だけにとどめておいた方が良いでしょう。

### ●G (en)

楽譜データを演奏コードに変換します。ノイズのセット、エンベロープの指定などは、このコマンドの実行前に行っておかなければなりません。

### ●Q (uit)

ジェネレータ・コマンド・モードを抜けて、システム・コマンド・モードに戻ります。

このジェネレータで生成される演奏コードは完全にリロケータブルなもので、\$1800番地から生成されます。演奏開始はBASICからの場合、EXEC &H1800で行ない、EXEC &H1802で停止します。どのような場合でも演奏が停止したらEXEC &H1802を実行してください。そうしないとインターバル・タイマーがONになったままになってしまい、カセットのロード、セーブができなくなります。

## プログラムについて

サウンド・エディタは、メイン・プログラム (記述言語: BASIC)、シーケンサ (アセンブラ)、ディスプレイ (Kコンパイラ)、エディタ (K)、ジェネレータ (K) という5つのプログラムで構成されています。

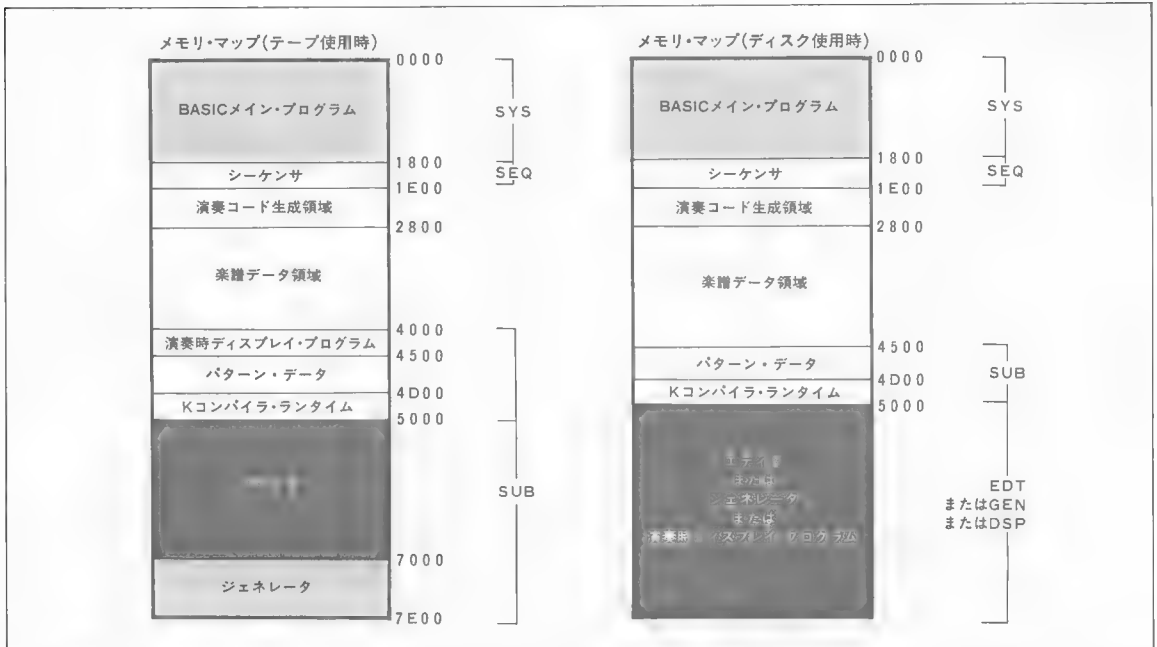
Kコンパイラによって書かれた3つのプログラムは、今回紙面の都合でオブジェクトで発表させていただきます。参考までに図1にメモリ・マップを示しておきました。

ところでこのプログラムをディスクで使っている場合には、楽譜データの領域を1Kバイト程度広く使うことができます。メイン・プログラムの10800行を次に示すように変更してください。

LSTEND=&H3FFF→LSTEND=&H44FF

ただしこの変更をしたプログラムで、カセットにセーブ

図 1



した楽譜データは、カセット版ではロードすることができないので注意してください。

## 終わりに

最近ローコストのパソコンが各社から発表されていますが、PSGを標準実装した形で発売される傾向があるようです。ゲームなどでも、グラフィックの美しさやキャラクターのかわいさなどの他に、サウンドが面白さを決定する要因になってくるでしょう。こういったハードウェアの機能をフルに活かした質の高いソフトが現われて欲しいものです。

### リスト 1 サウンド・エディタ・BASICメイン・プログラム

```

10000 '*
10100 '* program Sound Editing System v
version 2.0
10200 '* date 11-MAR-83
10300 '* author Y,Shirai
10400 '*
10500 CLEAR 80,&H17FF:DEFINT A-Z
10600 WIDTH 80,25:CONSOLE 0,25
10700 SEDADR=&H1800:EDTADR=&H5000:GENADR=
&H7000:DSFADR=&H4000
10800 CODTOP=&H1E00:CODEND=&H27FF:LSTTOP=
&H2800:LSTEND=&H3FFF
10900 COAREA=&H4500
11000
11100 FOR I=0 TO 3
11200 COLOR I*2+1
11300 LOCATE 0,I+8:PRINT"■":LOCATE 56,I+8
:PRINT"■"
11400 NEXT
11500 LOCATE 5,8:PRINT"Welcome to Sound
Editing System version 2.0"
11600 LOCATE 5,9:PRINT"Copyright by Comp
ac Inc."
11700 LOCATE 5,10:PRINT"This is an applic
ation program for FM-7 and FM-8"
11800 LOCATE 5,11:PRINT"with a Programmab
le Sound Generator Unit."
11900
12000 IF PEEK(&H1800)=&H20 AND PEEK(&H4D0
0)=&H7E THEN 13100
12100 LOCATE 0,13:PRINT"Reading Subsystem
..."
12200 LOADM"SEQ"
12300 LOADM"SUB"
12400 FOR I=0 TO 5:POKE CODTOP+I,0:NEXT
12500 R=0:D=&H0000:GOSUB 21300
12600 R=1:D=CODTOP:GOSUB 21300

```

```

12700 R=2:D=CODEND:GOSUB 21300
12800 R=3:D=LSTTOP:GOSUB 21300
12900 R=4:D=LSTEND:GOSUB 21300
13000 R=5:D=CODTOP:GOSUB 21300
13100 ON ERROR GOTO 19600
13200
13300 COLOR7:LOCATE 0,0
13400 PRINT"COMMAND [SYS]: E(edit,G(en,P(1
ay,D(ir,L(oad,S(ave,R(ead,W(rite,K(ill,N(
ew, ?";
13500 CM$=INKEY$:IF CM$="" THEN 13500
13600 PRINT CHR$(12)
13700 IF CM$="E" THEN GOSUB 14800
13800 IF CM$="G" THEN GOSUB 15100
13900 IF CM$="P" THEN GOSUB 15400
14000 IF CM$="D" THEN GOSUB 15700
14100 IF CM$="L" THEN GOSUB 16500
14200 IF CM$="S" THEN GOSUB 17600
14300 IF CM$="K" THEN GOSUB 18100
14400 IF CM$="N" THEN GOSUB 18700
14500 IF CM$="R" THEN GOSUB 16100
14600 IF CM$="W" THEN GOSUB 17200
14700 GOTO 13200
14800
14900 EXEC EDTADR
15000 RETURN
15100
15200 EXEC GENADR
15300 RETURN
15400
15500 EXEC DSFADR
15600 RETURN
15700
15800 GOSUB 19200
15900 FILES FDS
16000 RETURN
16100

```

```

16200 GOSUB 19200
16300 LOADM FD$
16400 RETURN
16500
16600 MS=1:GOSUB 20600
16700 IF R$<>"Y" THEN 17100
16800 GOSUB 19200
16900 LOADM FD$
17000 R=0:D=&H0001:GOSUB 21300
17100 RETURN
17200
17300 R=5:GOSUB 21700
17400 ADRTOP=SEQADR:ADREND=D
17500 GOTO 17800
17600
17700 ADRTOP=LSITOP:ADREND=LSITEND
17800 GOSUB 19200
17900 SAVEM FD$,ADRTOP,ADREND,ADRTOP
18000 RETURN
18100
18200 GOSUB 19200
18300 MS=3:GOSUB 20600
18400 IF R$<>"Y" THEN 18600
18500 KILL FD$
18600 RETURN
18700
18800 MS=1:GOSUB 20600
18900 IF R$<>"Y" THEN 19100
19000 R=0:D=&H0000:GOSUB 21300
19100 RETURN
19200

```

```

19300 INPUT "What File";FD$
19400 FD$=FD$
19500 RETURN
19600
19700 IF ERR=64 THEN 20100
19800 PRINT CHR$(7);
19900 PRINT "ERROR #";ERR
20000 RESUME 13200
20100 MS=2:GOSUB 20600
20200 IF R$<>"Y" THEN 20500
20300 KILL FD$
20400 RESUME
20500 RESUME 13200
20600
20700 IF MS=1 THEN PRINT "Through away cur
rent buffer?";
20800 IF MS=2 THEN PRINT "Through away old
"+FD$+"?";
20900 IF MS=3 THEN PRINT "Kill "+FD$+"?";
21000 R$=INKEY$:IF R$="" THEN 21000
21100 PRINT R$
21200 RETURN
21300
21400 DH=INT(D/256):DL=D MOD 256
21500 POKE COAREA+R*2,DH:POKE COAREA+R*2+
1,DL
21600 RETURN
21700
21800 D=PEEK(COAREA+R*2)*256+PEEK(COAREA+
R*2+1)
21900 RETURN

```

## リスト 2 サウンド・エディタ(シーケンサ)アセンブル・リスト

				TTL	MUSIC SEQUENCER
				OPT	OBJ
10000					
10050					
10100				*	
10150				*	program MUSIC SEQUENCER 2.0
10200				*	date 14-MAR-83
10250				*	author Y,SHIRAI
10300				*	
10350				*	This is an application program for
10400				*	FM-7 and FM-8 with a PSG unit.
10450				*	
10500	1800			ORG	\$1800 position independent
10550	1800	20	0E	1810 ENTRY1 BRA	START to start music
10600	1802	20	64	1868 ENTRY2 BRA	STOP to stop music
10650				*	
10700				*	SYSTEM MEMORY MAP
10750				*	
10800	1804	FFF6		FIRVCT FDB	\$FFF6 FIRQ vector
10850	1806	01E0		FIRJMP FDB	\$01E0 FIRQ jump address
10900	1808	FD05		SUBINT FDB	\$FD05 sub-system interface address
10950	180A	FC80		CMMBUF FDB	\$FC80 communication buffer address
11000	180C	FD0D		PSGADR FDB	\$FD0D PSG address
11050				*	
11100				*	CONSTANT
11150				*	
11200		0000		SOPRAN EQU	0
11250		0001		ALTO EQU	1
11300		0002		TENDR EQU	2
11350				*	
11400				*	GLOBAL VARIABLE
11450				*	
11500	180E	0600		GLOBAL FDB	\$1E00-ENTRY1
11550				*	
11600		0000		SEQSTR EQU	0 start of sequence
11650		0006		SEQPNT EQU	6 sequential pointer
11700		000C		COUNT EQU	12 delay counter
11750		0012		OCTAVE EQU	18 octave resister
11800		0018		SCALE EQU	24 scale resister
11850		001E		REPEAT EQU	30 repeat resister
11900		0024		RPTCNT EQU	36 repeat counter
11950		0027		TEMPO EQU	39 tempo resister
12000		0029		WFORM EQU	41 envelope wave form
12050		002A		STATUS EQU	42 status resister
12100		002B		PART EQU	43 part resister
12150		002C		CTRL EQU	44 control resister
12200				*	
12250				*	START MUSIC
12300				*	
12350	1810			START EQU	*
12400				*	

```

12450 1810 34 56 PSHS D,X,U
12500 1812 1A 50 ORCC ##50 mask IRQ&FIRO
12550 1814 EC 8C F7 LDD GLOBAL,PCR set global base
12600 1817 33 8C E6 LEAU ENTRY1,PCR
12650 181A 33 CB LEAU D,U
12700 181C 17 03DD 1BFC LBSR RESPSG reset PSG
12750 181F 6F CB 2A CLR STATUS,U clear halt flags
12800 1822 1F 30 TFR U,D
12850 1824 AE C4 LDX SEQSTR+SOPRAN*2,U set sequential point
12900 1826 30 8B LEAX D,X and repeat resister
12950 1828 AF 46 STX SEQPNT+SOPRAN*2,U
13000 182A AF CB 1E STX REPEAT+SOPRAN*2,U
13050 182D AE 42 LDX SEQSTR+ALTO*2,U
13100 182F 30 8B LEAX D,X
13150 1831 AF 48 STX SEQPNT+ALTO*2,U
13200 1833 AF CB 20 STX REPEAT+ALTO*2,U
13250 1836 AE 44 LDX SEQSTR+TENOR*2,U
13300 1838 30 8B LEAX D,X
13350 183A AF 4A STX SEQPNT+TENOR*2,U
13400 183C AF CB 22 STX REPEAT+TENOR*2,U
13450 183F CC 00FF LDD ##00FF
13500 1842 ED 4C STD COUNT+SOPRAN*2,U init delay counter
13550 1844 ED 4E STD COUNT+ALTO*2,U
13600 1846 ED CB 10 STD COUNT+TENOR*2,U
13650 1849 ED CB 27 STD TEMPO,U init tempo resister
13700 184C 86 01 LDA #1 init repeat counter
13750 184E A7 CB 24 STA RPTCNT+SOPRAN,U
13800 1851 A7 CB 25 STA RPTCNT+ALTO,U
13850 1854 A7 CB 26 STA RPTCNT+TENOR,U
13900 1857 6F CB 2C CLR CTRL,U set tone to disable
13950 185A 30 8D 0026 LEAX PLAY,PCR set FIRO vector
14000 185E AF 9C A3 STX [FIRVCT,PCR]
14050 1861 17 032A 1B8E LBSR INTON interval on
14100 1864 1C AF ANDCC ##AF clear IRQ&FIRO mask
14150 1866 35 D6 PULS D,X,U,PC
14200 *
14250 * STOP MUSIC
14300 *
14350 1868 STOP EQU *
14400 *
14450 1868 34 56 PSHS D,X,U
14500 186A 1A 50 ORCC ##50 mask IRQ&FIRO
14550 186C EC 8C 9F LDD GLOBAL,PCR set global base
14600 186F 33 8C 8E LEAU ENTRY1,PCR
14650 1872 33 CB LEAU D,U
14700 1874 17 0385 1BFC LBSR RESPSG reset PSG
14750 1877 17 0348 1BC2 LBSR INTOFF interval off
14800 187A AE 8C 89 LDX FIRJMP,PCR reset FIRO vector
14850 187D AF 9C 84 STX [FIRVCT,PCR]
14900 1880 1C AF ANDCC ##AF clear IRQ&FIRO mask
14950 1882 35 D6 PULS D,X,U,PC
15000 *
15050 * PLAY MUSIC
15100 *
15150 1884 PLAY EQU *
15200 0008 PLYWRK EQU 8
15250 *
15300 1884 34 F6 PSHS D,X,Y,U,PC
15350 1886 EC 8C 85 LDD GLOBAL,PCR set global base
15400 1889 33 8D FF73 LEAU ENTRY1,PCR
15450 188D 33 CB LEAU D,U
15500 188F AE 8D FF77 LDX CMMBUF,PCR
15550 1893 A6 01 LDA 1,X check interval flag
15600 1895 85 20 BITA ##20 interval?
15650 1897 27 19 18B2 BEQ YALP NO return
15700 1899 84 DF ANDA ##DF clear interval flag
15750 189B A7 01 STA 1,X
15800 189D 86 00 LDA #SOPRAN
15850 189F A7 CB 2B STA PART,U
15900 18A2 8D 16 18BA BSR RUN play SOPRAND
15950 18A4 86 01 LDA #ALTO
16000 18A6 A7 CB 2B STA PART,U
16050 18A9 8D 0F 18BA BSR RUN play ALTO
16100 18AB 86 02 LDA #TENOR
16150 18AD A7 CB 2B STA PART,U
16200 18B0 8D 08 18BA BSR RUN play TENOR
16250 18B2 AE 8D FF50 YALP LDX FIRJMP,PCR
16300 18B6 AF 68 STX PLYWRK,S
16350 18B8 35 F6 PULS D,X,Y,U,PC
16400 *
16450 * subroutine RUN
16500 * date 15-FEB-82
16550 * parameter list
16600 * U := < global_base >
16650 * all resisters are reserved
16700 *
16750 18BA RUN EQU *

```

```

16800      *
16850      18BA 34 36      PSHS D,X,Y
16900      18BC A6 C8 2B LDA PART,U check halt flag
16950      18BF C6 01      LDB ##1
17000      18C1 4A      RUN1 DECA
17050      18C2 2B 03 18C7 BMI RUN2
17100      18C4 58      ASLB
17150      18C5 20 FA 18C1 BRA RUN1
17200      18C7 E4 C8 2A RUN2 ANDB STATUS,U halt?
17250      18CA 26 75 1941 BNE NUR YES return
17300      18CC 30 4C      LEAX COUNT,U update delay counter
17350      18CE A6 C8 2B LDA PART,U
17400      18D1 48      ASLA
17450      18D2 30 86      LEAX A,X
17500      18D4 EC 84      LDD ,X
17550      18D6 A3 C8 27 SUBD TEMPO,U
17600      18D9 ED 84      STD ,X
17650      18DB 4D      TSTA count up?
17700      18DC 26 63 1941 BNE NUR NO return
17750      18DE 30 46      LEAX SEQPNT,U
17800      18E0 A6 C8 2B LDA PART,U
17850      18E3 48      ASLA X :=
17900      18E4 30 86      LEAX A,X < sequential_pointer_address >
17950      18E6 C6 7F      LDB #127 set loop counter
18000      18EB 10AE 84 EXEC LDY ,X get instruction code
18050      18EB A6 A0      LDA ,Y+
18100      18ED 10AF 84      STY ,X
18150      18F0 81 10      CMPA ##10 case CODE
18200      18F2 24 05 18F9 BHS CODE1 $0X:HALT
18250      18F4 8D 4D 1943 BSR HALT
18300      18F6 5F      CLRBB exist loop
18350      18F7 20 45 193E BRA NEXT
18400      18F9 81 20 CODE1 CMPA ##20 $1X:DELAY
18450      18FB 24 05 1902 BHS CODE2
18500      18FD 8D 59 1958 BSR DELAY
18550      18FF 5F      CLRBB exist loop
18600      1900 20 3C 193E BRA NEXT
18650      1902 81 30 CODE2 CMPA ##30 $2X:TONE
18700      1904 24 04 190A BHS CODE3
18750      1906 8D 6B 1973 BSR TONE
18800      1908 20 34 193E BRA NEXT
18850      190A 81 40 CODE3 CMPA ##40 $3X:LEVEL
18900      190C 24 05 1913 BHS CODE4
18950      190E 17 00E7 19FB LBSR LEVEL
19000      1911 20 2B 193E BRA NEXT
19050      1913 81 50 CODE4 CMPA ##50 $4X:ENVELOPE
19100      1915 24 05 191C BHS CODE5
19150      1917 17 00FB 1A15 LBSR ENVELO
19200      191A 20 22 193E BRA NEXT
19250      191C 81 60 CODE5 CMPA ##60 $5X:SWITCH
19300      191E 24 05 1925 BHS CODE6
19350      1920 17 00FB 1A1E LBSR SWITCH
19400      1923 20 19 193E BRA NEXT
19450      1925 81 70 CODE6 CMPA ##70 $6X:OPERAND
19500      1927 24 05 192E BHS CODE7
19550      1929 17 01AA 1AD6 LBSR OPERAN
19600      192C 20 10 193E BRA NEXT
19650      192E 81 80 CODE7 CMPA ##80 $7X:BRANCH
19700      1930 24 05 1937 BHS CODE8
19750      1932 17 01FA 1B2F LBSR BRANCH
19800      1935 20 07 193E BRA NEXT
19850      1937 81 90 CODE8 CMPA ##90 $8X:LOOP
19900      1939 24 03 193E BHS NEXT
19950      193B 17 021D 1B5B LBSR LOOP
20000      193E 5A      NEXT DECB decrement loop counter
20050      193F 2A A7 18E8 BPL EXEC
20100      1941 35 B6 NUR PULS D,X,Y,PC
20150      *
20200      * subroutine HALT
20250      * date 15-FEB-82
20300      * parameter list
20350      * U := < global base >
20400      * all registers are reserved
20450      *
20500      1943 HALT EQU *
20550      *
20600      1943 34 06      PSHS D
20650      1945 A6 C8 2B LDA PART,U set halt flag
20700      1948 C6 01      LDB ##1
20750      194A 4A      HALT1 DECA
20800      194B 2B 03 1950 BMI HALT2
20850      194D 58      ASLB
20900      194E 20 FA 194A BRA HALT1
20950      1950 EA C8 2A HALT2 ORB STATUS,U
21000      1953 E7 C8 2A STB STATUS,U
21050      1956 35 B6 PULS D,PC
21100      *

```

```

21150      *      subroutine DELAY
21200      *      date      15-FEB-82
21250      *      parameter list
21300      *      A := < instruction_code :$10..$1F >
21350      *      U := < global_base >
21400      *      all registers are reserved
21450      *
21500      1958      DELAY EQU      *
21550      0000      DLYCOD EQU    0
21600      *
21650      1958 34 36      PSHS    D,X,Y
21700      195A 30 4C      LEAX    COUNT,U
21750      195C A6 C8 2B   LDA     PART,U
21800      195F 48        ASLA
21850      1960 30 86      LEAX    A,X
21900      1962 31 8D 03BF LEAY    LENDAT,PCR caliculate
21950      1966 A6 E4      LDA     DLYCOD,S length-data address
22000      1968 B4 0F      ANDA    #$0F
22050      196A 48        ASLA
22100      196B EC A6      LDD     A,Y
22150      196D E3 84      ADDD    ,X
22200      196F ED 84      STD     ,X      set delay counter
22250      1971 35 B6      PULS    D,X,Y,PC
22300      *
22350      *      subroutine TONE
22400      *      date      14-MAR-83
22450      *      parameter list
22500      *      A := < instruction_code : $20..$2C >
22550      *      U := < global_base >
22600      *      all registers are reserved
22650      *
22700      1973      TONE EQU      *
22750      0001      TONCOD EQU    1
22800      0000      TONWRK EQU    0
22850      *
22900      1973 34 36      PSHS    D,X,Y
22950      1975 32 7F      LEAS    -1,S
23000      1977 81 2C      CMPA    #$2C
23050      1979 22 79 19F4 BHI     ENOT
23100      197B 31 C8 18   LEAY    SCALE,U
23150      197E E6 C8 2B   LDB     PART,U
23200      1981 58        ASLB
23250      1982 31 A5      LEAY    B,Y
23300      1984 B4 0F      ANDA    #$0F
23350      1986 A7 21      STA     1,Y      set scale resister
23400      1988 6F A4      CLR     ,Y
23450      198A 5D        TSTB
23500      198B 26 01 19BE BNE     *+3
23550      198D 5C        INCB
23600      198E E7 E4      STB     TONWRK,S
23650      1990 E4 C8 2C   ANDB    CTRL,U tone flag=0?
23700      1993 27 5F 19F4 BEQ     ENOT YES return
23750      1995 81 0C      CMPA    #$0C rest?
23800      1997 26 09 19A2 BNE     T01
23850      1999 86 07      LDA     #7
23900      199B 17 0289 1C27 LBSR   REDPSG
23950      199E EA E4      ORB     TONWRK,S tone off
24000      19A0 20 09 19AB BRA     T02
24050      19A2 B6 07      LDA     #7
24100      19A4 17 0280 1C27 LBSR   REDPSG
24150      19A7 63 E4      COM     TONWRK,S
24200      19A9 E4 E4      ANDB    TONWRK,S tone on
24250      19AB 17 0261 1C0F T02   LBSR   WRTPSG
24300      19AE E6 C8 2B   LDB     PART,U
24350      19B1 58        ASLB
24400      19B2 30 C8 12   LEAX    OCTAVE,U
24450      19B5 EC 85      LDD     B,X      caliculate
24500      19B7 30 8D 02B4 LEAX    SCLDAT,PCR scale-data address
24550      19BB B6 1A      LDA     #26
24600      19BD 3D        MUL
24650      19BE 30 8B      LEAX    D,X
24700      19C0 A6 61      LDA     TONCOD,S
24750      19C2 84 0F      ANDA    #$0F
24800      19C4 48        ASLA
24850      19C5 EC 86      LDD     A,X
24900      19C7 A7 E4      STA     TONWRK,S
24950      19C9 A6 C8 2B   LDA     PART,U set tone resister
25000      19CC 48        ASLA
25050      19CD 17 023F 1C0F LBSR   WRTPSG
25100      19D0 E6 E4      LDB     TONWRK,S
25150      19D2 4C        INCA
25200      19D3 17 0239 1C0F LBSR   WRTPSG
25250      19D6 A6 61      LDA     TONCOD,S
25300      19D8 B1 2C      CMPA    #$2C
25350      19DA 27 18 19F4 BEQ     ENOT
25400      19DC A6 C8 2B   LDA     PART,U
25450      19DF C6 10      LDB     #$10

```



```

25500 19E1 4A      TON1  DECA
25550 19E2 2B      03    19E7  BMI      TON2
25600 19E4 58      ASLB
25650 19E5 20      FA    19E1  BRA      TON1
25700 19E7 E5      C8 2A  TON2  BITB  STATUS,U check envelope flag
25750 19EA 27      08    19F4  BEQ      ENOT
25800 19EC 86      0D      LDA      #13      if flag=1 then set trigge
25850 19EE E6      C8 29  LDB      WFORM,U
25900 19F1 17      021B 1C0F LBSR   WRTPSG
25950 19F4 32      61      ENOT  LEAS   1,S
26000 19F6 35      B6      PULS   D,X,Y,PC
26050
26100      *
26150      *      subroutine LEVEL
26200      *      date      17-FEB-82
26250      *      parameter list
26300      *      A := < instruction_code : $30..$3F >
26350      *      U := < global_base >
26400      *      all registers are reserved
26450      *
26450      19F8      LEVEL EQU      *
26500      0001      LEVCOD EQU    1
26550      0000      LEVWRK EQU    0
26600      *
26650 19F8 34      06      PSHS   D      set level
26700 19FA 32      7F      LEAS   -1,S
26750 19FC 86      08      LDA    #8
26800 19FE AB      C8 2B  ADDA   PART,U
26850 1A01 17      0223 1C27 LBSR   REDPSG
26900 1A04 C4      10      ANDB   #$10
26950 1A06 E7      E4      STB    LEVWRK,S
27000 1A08 E6      61      LDB    LEVCOD,S
27050 1A0A C4      0F      ANDB   #$0F
27100 1A0C EA      E4      ORB    LEVWRK,S
27150 1A0E 17      01FE 1C0F LBSR   WRTPSG
27200 1A11 32      61      LEAS   1,S
27250 1A13 35      B6      PULS   D,PC
27300
27350      *
27400      *      subroutine ENVELOPE
27450      *      date      15-FEB-82
27500      *      parameter list
27550      *      A := < instruction_code : $40..$4F >
27600      *      U := < global_base >
27650      *      all registers are reserved
27700      *
27700      1A15      ENVELO EQU    *
27750      *
27800 1A15 34      02      PSHS   A      set envelope wave form
27850 1A17 84      0F      ANDA   #$0F
27900 1A19 A7      C8 29  STA    WFORM,U
27950 1A1C 35      82      PULS   A,PC
28000
28050      *
28100      *      subroutine SWITCH
28150      *      date      14-MAR-83
28200      *      parameter list
28250      *      A := < instruction_code : $50..$55 >
28300      *      U := < global_base >
28350      *      all registers are reserved
28400      *
28400      1A1E      SWITCH EQU    *
28450      0001      SWICOD EQU    1
28500      0000      SWIWRK EQU    0
28550      *
28600 1A1E 34      06      PSHS   D
28650 1A20 32      7F      LEAS   -1,S
28700 1A22 86      07      LDA    #7
28750 1A24 17      0200 1C27 LBSR   REDPSG
28800 1A27 A6      61      LDA    SWICOD,S
28850 1A29 B1      50      CMPA   #$50      case CODE
28900 1A2B 26      11      1A3E  BNE    SWIT0  $50:tone on
28950 1A2D A6      C8 2B  LDA    PART,U
29000 1A30 C6      01      LDB    #1
29050 1A32 17      009A 1ACF LBSR   BITSEL
29100 1A35 EA      C8 2C  ORB    CTRL,U      set tone flag
29150 1A3B E7      C8 2C  STB    CTRL,U
29200 1A3B 16      00BD 1ACB LBRA   HCTIWS
29250 1A3E B1      51      SWIT0  CMPA   #$51      $51:tone off
29300 1A40 26      11      1A53  BNE    SWIN1
29350 1A42 A6      C8 2B  LDA    PART,U
29400 1A45 C6      01      LDB    #1
29450 1A47 17      0085 1ACF LBSR   BITSEL
29500 1A4A 53      COMB
29550 1A4B EA      C8 2C  ANDB   CTRL,U      clear tone flag
29600 1A4E E7      C8 2C  STB    CTRL,U
29650 1A51 20      78      1ACB  BRA    HCTIWS
29700 1A53 B1      52      SWIN1  CMPA   #$52      $52:noise on
29750 1A55 26      14      1A6B  BNE    SWIN0
29800 1A57 E7      E4      STB    SWIWRK,S

```

```

29850 1A59 A6 C8 2B LDA PART,U
29900 1A5C C6 08 LDB #$8
29950 1A5E 8D 6F 1ACF BSR BITSEL
30000 1A60 C8 FF EORB #$FF
30050 1A62 E4 E4 ANDB SWIWRK,S
30100 1A64 86 07 LDA #7
30150 1A66 17 01A6 1C0F LBSR WRTPSG
30200 1A69 20 60 1ACB BRA HCTIWS
30250 1A6B 81 53 SWIN01 CMPA #$53 $53:noise off
30300 1A6D 26 12 1AB1 BNE SWIVE
30350 1A6F E7 E4 STB SWIWRK,S
30400 1A71 A6 C8 2B LDA PART,U
30450 1A74 C6 08 LDB #$8
30500 1A76 8D 57 1ACF BSR BITSEL
30550 1A78 EA E4 ORB SWIWRK,S
30600 1A7A 86 07 LDA #7
30650 1A7C 17 0190 1C0F LBSR WRTPSG
30700 1A7F 20 4A 1ACB BRA HCTIWS
30750 1A81 86 08 SWIVE LDA #8
30800 1A83 AB C8 2B ADDA PART,U
30850 1A86 17 019E 1C27 LBSR REDPSG
30900 1A89 A6 61 LDA SWICOD,S
30950 1A8B 81 54 CMPA #$54 case CODE
31000 1A8D 26 1D 1AAC BNE SWIVO $54:volume on , envelope off
31050 1A8F C4 EF ANDB #$EF
31100 1A91 86 08 LDA #8
31150 1A93 AB C8 2B ADDA PART,U
31200 1A96 17 0176 1C0F LBSR WRTPSG
31250 1A99 A6 C8 2B LDA PART,U reset envelope flag
31300 1A9C C6 EF LDB #$EF
31350 1A9E 4A SWVE1 DECA
31400 1A9F 2B 03 1AA4 BMI SWVE2
31450 1AA1 5B ASLB
31500 1AA2 20 FA 1A9E BRA SWVE1
31550 1AA4 E4 C8 2A SWVE2 ANDB STATUS,U
31600 1AA7 E7 C8 2A STB STATUS,U
31650 1AAA 20 1F 1ACB BRA HCTIWS
31700 1AAC 81 55 SWIVO1 CMPA #$55 $55:volume off , envelope on
31750 1AAE 26 1B 1ACB BNE HCTIWS
31800 1AB0 CA 10 ORB #$10
31850 1AB2 86 08 LDA #8
31900 1AB4 AB C8 2B ADDA PART,U
31950 1AB7 17 0155 1C0F LBSR WRTPSG
32000 1ABA A6 C8 2B LDA PART,U set envelope flag
32050 1ABD C6 10 LDB #$10
32100 1ABF 4A SWVO1 DECA
32150 1AC0 2B 03 1AC5 BMI SWVO2
32200 1AC2 5B ASLB
32250 1AC3 20 FA 1ABF BRA SWVO1
32300 1AC5 EA C8 2A SWVO2 ORB STATUS,U
32350 1AC8 E7 C8 2A STB STATUS,U
32400 1ACB 32 61 HCTIWS LEAS 1,S
32450 1ACD 35 86 FULS D,PC
32500 *
32550 1ACF 4A BITSEL DECA
32600 1AD0 2B 03 1AD5 BMI LESTIB
32650 1AD2 5B ASLB
32700 1AD3 20 FA 1ACF BRA BITSEL
32750 1AD5 39 LESTIB RTS
32800 *
32850 * subroutine OPERAND
32900 * date 17-FEB-82
32950 * parameter list
33000 * A := < instruction_code : $60,$61,$62,$63,$64
33050 * U := < global_base >
33100 * all registers are reserved
33150 *
33200 1AD6 OPERAN EQU *
33250 0006 OPEGB EQU 6
33300 *
33350 1AD6 34 76 PSHS D,X,Y,U
33400 1ADB 30 46 LEAX SECPNT,U
33450 1ADA E6 C8 2B LDB PART,U
33500 1ADD 5B ASLB
33550 1ADE 30 85 LEAX B,X
33600 1AE0 EE 84 LDU ,X
33650 1AE2 10AE C1 LDY ,U++ Y := < operand >
33700 1AE5 EF 84 STU ,X
33750 1AE7 EE 66 LDU OPEGB,S
33800 1AE9 81 60 CMPA #$60 case CODE
33850 1AEB 26 06 1AF3 BNE SETOCT $60:set tempo resister
33900 1AED 10AF C8 27 STY TEMPO,U
33950 1AF1 20 3A 1B2D BRA NAREPO
34000 1AF3 81 61 SETOCT CMPA #$61 $61:set octave resister
34050 1AF5 26 0C 1B03 BNE NISFRQ
34100 1AF7 30 C8 12 LEAX OCTAVE,U
34150 1AFA A6 C8 2B LDA PART,U

```

```

34200 1AFD 48          ASLA
34250 1AFE 10AF 86     STY      A,X
34300 1B01 20 2A      1B2D    BRA      NAREFO
34350 1B03 81 62      NISFRQ CMPA    ##62    $62:set noise frequency
34400 1B05 26 09      1B10    BNE     ENVFRQ  f=1.2288E+06/16*d
34450 1B07 1F 20      TFR      Y,D
34500 1B09 86 06      LDA      #6
34550 1B0B 17 0101 1C0F    LBSR    WRTPSG
34600 1B0E 20 1D      1B2D    BRA      NAREFO
34650 1B10 81 63      ENVFRQ CMPA    ##63    $63:set envelope frequency
34700 1B12 26 12      1B26    BNE     OPTION  f=1.2288E+06/256*d
34750 1B14 1F 20      TFR      Y,D
34800 1B16 86 0B      LDA      #11
34850 1B18 17 00F4 1C0F    LBSR    WRTPSG
34900 1B1B 1F 20      TFR      Y,D
34950 1B1D 1F 89      TFR      A,B
35000 1B1F 86 0C      LDA      #12
35050 1B21 17 00EB 1C0F    LBSR    WRTPSG
35100 1B24 20 07      1B2D    BRA      NAREFO
35150 1B26 81 64      OPTION CMPA    ##64    $64:option
35200 1B28 1F 20      TFR      Y,D
35250 1B2A 17 00E2 1C0F    LBSR    WRTPSG
35300 1B2D 35 F6      NAREFO PULS    D,X,Y,U,PC
35350
35400
35450
35500
35550
35600
35650
35700
35750
35800
35850
35900
35950 1B2F 34 76      PSHS     D,X,Y,U
36000 1B31 30 46      LEAX     SEQPNT,U
36050 1B33 E6 CB 2B    LDB      PART,U
36100 1B36 58          ASLB
36150 1B37 30 85      LEAX     B,X
36200 1B39 EE 84      LDU      ,X
36250 1B3B 10AE C1     LDY      ,U++    Y := < operand >
36300 1B3E EF 84      STU      ,X
36350 1B40 EE 66      LDU      BRAGB,S
36400 1B42 A6 E4      LDA      BRACOD,S
36450 1B44 84 0F      ANDA     #$0F
36500 1B46 27 0A      1B52    BEQ     BRAALW
36550 1B48 E6 CB 2B    LDB      PART,U
36600 1B4B 33 CB 24    LEAU     RPTCNT,U
36650 1B4E A1 C5      CMPA     B,U
36700 1B50 26 07      1B59    BNE     HCNARB
36750 1B52 EC 84      BRAALW   LDD     ,X
36800 1B54 31 AB      LEAY     D,Y
36850 1B56 10AF 84     STY      ,X
36900 1B59 35 F6      HCNARB   PULS    D,X,Y,U,PC
36950
37000
37050
37100
37150
37200
37250
37300
37350
37400 1B5B 34 76      LOOP     EQU     *
37450 1B5D E6 CB 2B    LDB      PART,U
37500 1B60 30 CB 24    LEAX     RPTCNT,U
37550 1B63 31 46      LEAY     SEQPNT,U
37600 1B65 33 CB 1E    LEAU     REPEAT,U
37650 1B68 30 85      LEAX     B,X
37700 1B6A 58          ASLB
37750 1B6B 31 A5      LEAY     B,Y
37800 1B6D 33 C5      LEAU     B,U
37850 1B6F 81 80      CMPA     ##80    case CODE
37900 1B71 26 0A      1B7D    BNE     ENDLOOP  $80:LOOP
37950 1B73 C6 01      LDB      #1      reset repeat counter
38000 1B75 E7 84      STB      ,X      set repeat resister
38050 1B77 EC A4      LDD      ,Y
38100 1B79 ED C4      STD      ,U
38150 1B7B 20 0F      1B8C    BRA      POOL
38200 1B7D 81 81      ENDLOOP CMPA    ##81    $81:END_LOOP
38250 1B7F 26 0B      1B8C    BNE     POOL
38300 1B81 E6 84      LDB      ,X      increment repeat resister
38350 1B83 5C          INCB
38400 1B84 C4 0F      ANDB     #$0F
38450 1B86 E7 84      STB      ,X

```

```

38500 1888 EC C4          LDD    ,U      repeat
38550 188A ED A4          STD     ,Y
38600 188C 35 F6          POOL    PULS   D,X,Y,U,PC
38650                      *
38700                      *      subroutine INTERVAL ON
38750                      *      date      25-FEB-82
38800                      *      all registers are reserved
38850                      *      IRQ&FIQ must be masked!
38900                      *
38950                      188E .      INTON  EQU      *
39000                      *
39050 188E 34 16          PSHS    D,X
39100 1890 AE 8D FC76     LDX     CMMBUF,PCR
39150 1894 8D 4E 18E4     BSR     STPSUB
39200 1896 86 3E          LDA     ##3E      read timer
39250 1898 A7 02          STA     2,X      C
39300 189A 8D 5B 1BF7     BSR     RSTSUB
39350 189C 8D 46 18E4     BSR     STPSUB
39400 189E 86 3D          LDA     ##3D      set timer
39450 18A0 A7 02          STA     2,X      C
39500 18A2 86 19          LDA     ##19      set TC,T2,T2-D
39550 18A4 A7 03          STA     3,X      RS
39600 18A6 A6 04          LDA     4,X
39650 18AB 84 FB          ANDA    ##FB      interval mode
39700 18AA 8A 02          ORA     ##02      enable interval
39750 18AC A7 04          STA     4,X      TC
39800 18AE CC 0000        LDD     #0
39850 18B1 ED 0D          STD     13,X
39900 18B3 ED 88 11      STD     17,X
39950 18B6 CC 0001        LDD     #1
40000 18B9 ED 0F          STD     15,X      T2=$00000001 (20ms)
40050 18BB ED 88 13      STD     19,X      T2-D=$00000001
40100 18BE 8D 37 1BF7     BSR     RSTSUB
40150 18C0 35 96          PULS    D,X,PC
40200                      *
40250                      *      subroutine INTERVAL OFF
40300                      *      date      25-FEB-82
40350                      *      all registers are reserved
40400                      *      IRQ&FIQ must be masked!
40450                      *
40500                      18C2      INTOFF EQU      *
40550                      *
40600 18C2 34 12          PSHS    A,X
40650 18C4 AE 8D FC42     LDX     CMMBUF,PCR
40700 18C8 8D 1A 18E4     BSR     STPSUB
40750 18CA 86 3E          LDA     ##3E      read timer
40800 18CC A7 02          STA     2,X      C
40850 18CE 8D 27 1BF7     BSR     RSTSUB
40900 18D0 8D 12 18E4     BSR     STPSUB
40950 18D2 86 3D          LDA     ##3D      set timer
41000 18D4 A7 02          STA     2,X      C
41050 18D6 86 01          LDA     ##01      set TC
41100 18D8 A7 03          STA     3,X      RS
41150 18DA A6 04          LDA     4,X
41200 18DC 84 FD          ANDA    ##FD      disable interval
41250 18DE A7 04          STA     4,X      TC
41300 18E0 8D 15 1BF7     BSR     RSTSUB
41350 18E2 35 92          PULS    A,X,PC
41400                      *
41450 18E4 6D 9D FC20     STPSUB  TST     [SUBINT,PCR]
41500 18E8 2B FA 18E4     BMI     STPSUB  ready?
41550 18EA 86 80          LDA     ##80      stop sub-system
41600 18EC A7 9D FC18     STA     [SUBINT,PCR]
41650 18F0 6D 9D FC14     STPSU1  TST     [SUBINT,PCR] busy?
41700 18F4 2A FA 1BF0     BPL     STPSU1
41750 18F6 39            RTS
41800                      *
41850 18F7 6F 9D FC0D     RSTSUB  CLR     [SUBINT,PCR] restart sub-system
41900 18FB 39            RTS
41950                      *
42000                      *      subroutine RESET PSG
42050                      *      date      1-FEB-82
42100                      *      all registers are reserved
42150                      *
42200                      18FC      RESPSG EQU      *
42250                      *
42300 18FC 34 06          PSHS    D
42350 18FE 86 0F          LDA     #15      reset PSG registers
42400 1C00 C6 00          LDB     ##0
42450 1C02 8D 0B 1C0F     BSR     WRTPSG
42500 1C04 4A          DECA
42550 1C05 2A FB 1C02     BPL     RESPS1
42600 1C07 86 07          LDA     #7
42650 1C09 C6 FF          LDB     ##FF
42700 1C0B 8D 02 1C0F     BSR     WRTPSG

```

```

42750 1C0D 35 86          PULS  D,PC
42800
42850
42900
42950
43000
43050
43100
43150
43200          1C0F
43250
43300 1C0F 34 12          PSHS  A,X
43350 1C11 AE 8D FBF7     LDX   PSGADR,PCR
43400 1C15 A7 01          STA   1,X
43450 1C17 B6 03          LDA   ##3      ratch address
43500 1C19 A7 84          STA   ,X
43550 1C1B 6F 84          CLR   ,X      inactive
43600 1C1D E7 01          STB   1,X
43650 1C1F B6 02          LDA   ##2      write data
43700 1C21 A7 84          STA   ,X
43750 1C23 6F 84          CLR   ,X      inactive
43800 1C25 35 92          PULS  A,X,PC
43850
43900
43950
44000
44050
44100
44150
44200
44250
44300          1C27
44350
44400 1C27 34 12          PSHS  A,X
44450 1C29 AE 8D FBDF     LDX   PSGADR,PCR
44500 1C2D A7 01          STA   1,X
44550 1C2F B6 03          LDA   ##3      ratch address
44600 1C31 A7 84          STA   ,X
44650 1C33 6F 84          CLR   ,X      inactive
44700 1C35 B6 01          LDA   ##1      read data
44750 1C37 A7 84          STA   ,X
44800 1C39 E6 01          LDB   1,X
44850 1C3B 6F 84          CLR   ,X      inactive
44900 1C3D 35 92          PULS  A,X,PC
44950
45000
45050
45100          1C3F
45150
45200 1C3F 0922          SCLDAT EQU *
      1C41 089F          * OCTAVE 0
      1C43 0823          FDB    $0922,$089F,$0823,$07AE,$073F,$06D7
      1C45 07AE
      1C47 073F
      1C49 06D7
45250 1C4B 0675          FDB    $0675,$0618,$05C1,$056E,$0520,$04D6
      1C4D 0618
      1C4F 05C1
      1C51 056E
      1C53 0520
      1C55 04D6
45300 1C57 0000          FDB    $0000
45350
45400 1C59 0491          * OCTAVE 1
      1C5B 044F          FDB    $0491,$044F,$0411,$03D7,$03A0,$036C
      1C5D 0411
      1C5F 03D7
      1C61 03A0
      1C63 036C
45450 1C65 0338          FDB    $0338,$030C,$02E0,$02B7,$0290,$026B
      1C67 030C
      1C69 02E0
      1C6B 02B7
      1C6D 0290
      1C6F 026B
45500 1C71 0000          FDB    $0000
45550
45600 1C73 0248          * OCTAVE 2
      1C75 0228          FDB    $0248,$0228,$0209,$01EB,$01D0,$01B6
      1C77 0209
      1C79 01EB
      1C7B 01D0
      1C7D 01B6
45650 1C7F 019D          FDB    $019D,$01B6,$0170,$015C,$0148,$0136
      1C81 0186

```

	1C83	0170		
	1C85	015C		
	1C87	0148		
	1C89	0136		
45700	1C8B	0000		
45750			* FDB	\$0000
45800			* OCTAVE 3	
			FDB	\$0124,\$0114,\$0104,\$00F6,\$00EB,\$00DB
	1C8D	0124		
	1C8F	0114		
	1C91	0104		
	1C93	00F6		
	1C95	00EB		
	1C97	00DB		
45850	1C99	00CF	FDB	\$00CF,\$00C3,\$00BB,\$00AE,\$00A4,\$009B
	1C9B	00C3		
	1C9D	00BB		
	1C9F	00AE		
	1CA1	00A4		
	1CA3	009B		
45900	1CA5	0000		
45950			* FDB	\$0000
46000			* OCTAVE 4	
	1CA7	0092	FDB	\$0092,\$008A,\$0082,\$007B,\$0074,\$006D
	1CA9	008A		
	1CAB	0082		
	1CAD	007B		
	1CAF	0074		
	1CB1	006D		
46050	1CB3	0067	FDB	\$0067,\$0062,\$005C,\$0057,\$0052,\$004D
	1CB5	0062		
	1CB7	005C		
	1CB9	0057		
	1CBB	0052		
	1CBD	004D		
46100	1CBF	0000		
46150			* FDB	\$0000
46200			* OCTAVE 5	
	1CC1	0049	FDB	\$0049,\$0045,\$0041,\$003D,\$003A,\$0037
	1CC3	0045		
	1CC5	0041		
	1CC7	003D		
	1CC9	003A		
	1CCB	0037		
46250	1CCD	0034	FDB	\$0034,\$0031,\$002E,\$002B,\$0029,\$0027
	1CCF	0031		
	1CD1	002E		
	1CD3	002B		
	1CD5	0029		
	1CD7	0027		
46300	1CD9	0000		
46350			* FDB	\$0000
46400			* OCTAVE 6	
	1CDB	0025	FDB	\$0025,\$0022,\$0021,\$001F,\$001D,\$001B
	1CDD	0022		
	1CDF	0021		
	1CE1	001F		
	1CE3	001D		
	1CE5	001B		
46450	1CE7	001A	FDB	\$001A,\$0018,\$0017,\$0016,\$0015,\$0013
	1CE9	0018		
	1CEB	0017		
	1CED	0016		
	1CEF	0015		
	1CF1	0013		
46500	1CF3	0000		
46550			FDB	\$0000
46600			* LENGTH-DATA	
46650			* LENDAT EQU	*
46700	1CF5			
46750			* FDB	\$3000,\$2400,\$1800,\$1200
46800	1CF5	3000		
	1CF7	2400		
	1CF9	1800		
	1CFB	1200		
46850	1CFD	0C00	FDB	\$0C00,\$0900,\$0600,\$0480
	1CFF	0900		
	1D01	0600		
	1D03	0480		
46900	1D05	0300	FDB	\$0300,\$0240,\$0180,\$1000
	1D07	0240		
	1D09	0180		
	1D0B	1000		
46950	1D0D	0800	FDB	\$0800,\$0400,\$0200,\$0100
	1D0F	0400		
	1D11	0200		
	1D13	0100		
47000			* END	
47050				





## リスト 3

4A30	00	00	07	86	1A	01	00	00	00	00	00	00	00	00	00	00	00	:AB
4A40	00	00	E1	E1	E0	00	00	00	00	00	00	00	00	00	00	00	00	:A2
4A50	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4A60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4A70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4A80	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4A90	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4AA0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4AB0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4AC0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4AD0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4AE0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4AF0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
Sum:	61	B7	6B	67	FA	01	00	00	00	00	00	00	00	00	02	01	:B5	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
4B00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4B10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4B20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4B30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4B40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4B50	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4B60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4B70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4B80	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4B90	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4BA0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4BB0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4BC0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4BD0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4BE0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4BF0	00	06	00	03	00	01	00	A7	02	6B	00</							

4F70	80	B7	FD	05	B6	FD	05	2A	FB	35	B2	7F	FD	05	1C	00	:6A
4F80	39	34	05	C6	12	BD	59	8D	DD	CC	29	00	FD	FC	B2	80	:77
4F90	EA	8D	03	08	02	35	B5	6C	8D	B6	FC	B3	FA	FC	B0	F7	:7B
4FA0	FC	80	20	D7	34	35	6C	13	8D	C1	BD	BA	CC	04	02	FD	:BE
4FB0	FC	B2	CC	11	07	FD	FC	8A	8D	C1	BD	AA	F6	FC	03	26	:FF
4FC0	2A	F6	FC	84	27	12	5A	5A	5A	27	0D	10	BE	FC	B8	A6	:E3
4FD0	A0	27	05	A7	80	5A	26	F7	C6	0D	E7	B4	8D	B9	35	B5	:08
4FE0	8D	84	B6	0C	FD	FC	B2	20	92	00	00	4F	1F	8B	8D	A7	:FD
4FF0	B6	FD	01	B1	FF	27	06	7D	FD	0F	7E	D6	80	7E	94	8D	:5D
Sum:	DB	BB	8A	12	5E	4C	53	9A	32	BB	32	22	53	DC	F0	72	:6B
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5000	34	7F	C6	4C	1F	9B	CE	4C	00	CC	45	00	DD	00	CC	45	:9B
5010	80	DD	02	CC	46	00	DD	04	CC	47	80	DD	06	9E	00	CC	:52
5020	06	DD	08	9E	00	EC	0B	DD	0A	CC	00	0C	BD	4E	DE	CC	:01
5030	02	DD	0C	9E	00	EC	0B	DD	0A	CC	00	01	E7	80	CC	:48	
5040	00	80	E7	80	CC	00	02	E7	80	CC	00	81	E7	80	CC	:9C	
5050	03	E7	80	CC	00	82	E7	80	CC	00	00	E7	80	CC	00	:1E	
5060	E7	80	CC	00	00	E7	80	CC	00	00	E7	80	CC	00	00	:E7	
5070	80	CC	00	BB	E7	80	CC	00	78	E7	80	CC	00	00	E7	:80	
5080	CC	00	00	E7	80	CC	00	00	E7	80	CC	00	00	E7	80	:C9	
5090	00	FF	E7	80	CC	00	FF	E7	80	9F	CC	00	00	07	DD	:0E	
50A0	CC	00	07	DD	10	17	09	03	CC	00	06	DD	0E	CC	00	:16	
50B0	DD	12	1F	1A	39	37	1B	03	17	0C	9E	CC	00	00	36	:96	
50C0	CC	00	00	3A	06	37	3A	10	17	1E	39	32	64	BD	4D	:5A	
50D0	0B	43	4F	4D	4E	41	4E	44	20	5B	45	44	54	5D	3A	:20	
50E0	45	2B	64	69	74	2C	49	2B	6E	33	65	72	74	2C	44	:2B	
50F0	65	6C	65	74	65	2C	4E	2B	65	7B	74	20	70	61	67	:65	
Sum:	1C	B1	2C	E4	E5	06	00	9B	CE	9A	B0	21	4C	27	12	1B	:39
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5100	2C	50	2B	72	65	20	70	61	67	65	2C	4A	2B	75	3F	00	:2B
5110	20	70	61	67	65	2C	51	2B	75	69	74	2C	20	6D	00	:17	
5120	09	BB	DD	1F	DC	14	B3	00	00	27	F4	CC	00	00	36	:06	
5130	CC	00	00	3A	06	37	10	3A	10	17	1D	E9	32	64	CC	:00	
5140	4F	BD	4E	B6	DC	14	B3	00									

54B0	00	1C	17	17	3F	EC	62	DD	12	EC	62	34	06	17	16	7F	:FA
54C0	32	62	DD	22	DC	24	DD	12	17	16	F3	17	08	BB	39	EC	:71
54D0	62	B3	00	00	10	26	00	21	CC	00	01	34	06	17	FF	EF	:48
54E0	32	62	CC	00	02	34	06	17	FF	EC	62	B3	00	00	03	34	:2E
54F0	06	17	00	92	32	62	16	00	8C	EC	62	B3	00	04	10	2C	:F6
Sum:	DD	76	D9	49	32	16	CF	09	78	BE	16	0E	B5	A4	38	98	:18
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5500	00	B3	EC	62	34	06	17	06	09	32	62	CC	00	0A	36	06	:07
5510	DC	26	36	06	CC	00	64	36	06	CC	00	28	36	06	CC	00	:B7
5520	36	06	CC	00	03	36	06	CC	00	02	34	06	37	10	34	10	:DA
5530	37	10	34	10	37	10	34	10	37	10	34	10	37	10	34	10	:2C
5540	37	10	34	10	17	19	BD	32	6E	EC	62	B3	00	01	58	49	:5B
5550	9E	04	30	BB	36	10	CC	00	80	E7	D1	CC	00	80	36	06	:2F
5560	CC	00	00	34	06	37	10	34	10	17	08	BB	39	32	64	9E	:04
5570	4F	E6	06	36	06	9E	04	4F	E6	07	34	06	37	10	34	10	:1A
5580	17	08	31	32	64	39	EC	62	B3	00	00	10	26	00	21	CC	:13
5590	00	01	34	06	17	FF	38	32	62	CC	00	02	34	06	17	FF	:3E
55A0	2E	32	62	CC	00	03	34	06	17	FF	BD	32	62	16	00	8C	:F2
55B0	EC	62	B3	00	04	10	2C	00	8C	EC	62	34	06	17	05	52	:8A
55C0	32	62	CC	00	0A	36	06	CC	26	36	06	CC	00	64	36	06	:50
55D0	DC	28	36	06	CC	00	01	36	06	CC	00	03	36	06	CC	00	:20
55E0	02	34	06	37	10	34	10	37	10	34	10	37	10	34	10	37	:14
55F0	10	34	10	37	10	34	10	37	10	34	10	37	10	17	18	D6	:9F
Sum:	BA	48	EE	F5	08	33	CD	E7	F5	32	C4	4A	FD	92	7F	DE	:C5
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5600	EC	62	B3	00	01	58	49	9E	04	30	BB	36	10	CC	00	81	:63
5610	E7	D1	CC	00	01	36	06	CC	00	00	34	06	37	10	34	10	:D2
5620	17	07	31	32	64	9E	04	4F	E6	06	36	06	9E	04	4F	E6	:35
5630	07	34	06	37	10	34	10	17	07	7A	32	64	39	EC	62	B3	:04
5640	00	08	10	2C	00	82	CC	00	01	DD	2A	CC	00	03	36	06	:A5
5650	DC	2A	16	00	6D	CC	2A	34	06	17	04	B6	32	62	CC	00	:FA
5660	1E	36	06	CC	26	36	06	CC	00	64	36	06	CC	28	36	06	:44
5670	CC	00	01	36	06	CC	00	03	36	06	CC	00	02	34	06	37	:53
5680	10	34	10	37	10	34	10	37	10	34	10	37	10	34	10	37	:2C
5690	10	34	10	37	10	34	10	17	18	3A	32	6E	CC	00	82	9E	:D4
56A0	04	E7	06	EC	62	9E	04	E7	07	CC	00	B2	36	06	EC	62	:A7
56B0	34	06	37	10	34	10	17	06	FB	32	64	DC	2A	C3	00	01	:3D
56C0	DD	2A	A3	C4	2F	BF	33	42	39	EC	62	B3	00	08	10	2C	:EF
56D0	00	B2	CC	00	01	DD	2A	CC	00	03	36	06	CC	2A	16	00	:7F
56E0	6D	DC	2A	34	06	17	04	2A	32	62	CC	00	1E	36	06	CC	:88
56F0	26	36	06	CC	00	64	36	06	CC	28	36	06	CC	00	01	36	:11
Sum:	7F	E9	0F	D5	7B	BD	31	4C	9F	F3	97	C0	30	F2	CE	BD	:8D
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5700	06	CC	00	03	36	06	CC	00	02	34	06	37	10	34	10	37	:DB
5710	10	34	10	37	10	34	10	37	10	34	10	37	10	34	10	37	:2C
5720	10	34	10	17	17	AE	32	6E	CC	00	B3	9E	04	E7	06	EC	:9A
5730	62	9E	04	E7	07	CC	00	B3	9E	06	EC	62	34	06	37	10	:4C
5740	34	10	17	06	6F	32	64	DC	2A	C3	00	01	DD	2A	A3	C4	:9E
5750	2F	BF	33	42	39	EC	00	65	36	06	CC	00	28	36	06	CC	:D5
5760	00	77	36	06	CC	00	98	36	06	CC	00	01	36	06	CC	00	:28
5770	03	36	06	CC	00	02	34	06	37	10	34	10	37	10	34	10	:5D
5780	37	10	34	10	37	10	34	10	37	10	34	10	37	10	34	10	:2C
5790	17	17	41	32	6E	9E	04	4F	E6	0A	B3	00	84	10	26	00	:2D
57A0	0A	CC	00	00	9E	04	E7	0A	16	00	3F	CC	00	84	9E	04	:8C
57B0	E7	0A	CC	00	01	DD	2A	CC	00	03	36	06	CC	2A	16	00	:EC
57C0	23	DC	2A	34	06	17	03	4A	32	62	CC	00	84	36	06	CC	:B3
57D0	00	00	34	06	37	10	34	10	17	05	D9	32	64	DC	2A	C3	:19
57E0	00	01	DD	2A	A3	C4	2F	DF	32	62	CC	00	71	36	06	CC	:A0
57F0	CC	00	28	36	06	CC	02	7F	36	06	CC	00	98	36	06	CC	:25
Sum:	1C	F8	4E	2E	02	FA	EF	BC	96	DF	5B	60	E3	52	80	7F	:6B
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5800	00	01	36	06	CC	00	03	36	06	CC	00	02	34	06	37	10	:97
5810	34	10	37	10	34	10	37	10	34	10	37	10	34	10	37	10	:2C
5820	34	10	37	10	34	10	17	16	AE	32	6E	9E	04	4F	E6	06	:C1
5830	B3	00	85	10	26	00	0A	CC	00	00	9E	04	E7	0E	16	00	:2C
5840	3F	CC	00	85	9E	04	E7	0E	CC	00	01	DD	2A	CC	00	03	:1C
5850	36	06	CC	2A	16	00	23	DC	2A	34	06	37	10	34	10	37	:CA
5860	CC	00	85	36	06	CC	00	00	34	06	37	10	34	10	17	05	:3A
5870	43	32	64	DC	2A	C3	00	01	DD	2A	A3	C4	2F	D9	33	42	:8E
5880	39	CC	02	71	36	06	CC	00	28	36	06	CC	02	7F	36	06	:6D
5890	CC	00	98	36	06	CC	00	01	36	06	CC	00	03	36	06	CC	:80
58A0	00	02	34	06	37	10	34	10	37	10	34	10	37	10	34	10	:DD
58B0	37	10	34	10	37	10	34	10	37	10	34	10	37	10	17	16	:32
58C0	6E	9E	04	4F	E6	06	B3	00	86	10	26	00	0A	CC	00	00	:68
58D0	9E	04	E7	0E	16	00	3F	CC	00	86	9E	04	E7	0E	CC	00	:A1
58E0	01	DD	2A	CC	00	03	36	06	CC	2A	16	00	23	DC	2A	C3	:8C
58F0	06	17	02	1E	32	62	CC	00	B6	36	06	CC	00	00	34	06	:65
Sum:	BE	99	07	FB	16	18	5D	06	A0	C4	3E	38	49	6D	95	28	:37
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5900	37	10	34	10	17	04	AD	32	64	DC	2A	C3	00	01	DD	2A	:BA
5910	A3	C4	2F	D9	33	42	39	EC	00	00	36	06	CC	00	10	36	:37
5920	06	CC	02	7F	36	06	CC	00	17	36	06	CC	00	00	36	06	:B6
5930	CC	00	00	36	06	CC	00	00	34	06	37	10	34	10	37	10	:E2
5940	34	10	37	10	34	10	37	10	34	10	37	10	34	10	37	10	:2C
5950	34	10	17	15	7F	32	6E	EC	B3	00	00	10	26	00	0A	CC	:A0
5960	CC	00	00	9E	04	ED	8B	16	00	20	CC	00	87	9E	04	E7	:75
5970	08	EC	62	9E	04	ED	8B	10	CC	00	87	36	06	EC	62	34	:8E
5980	06	37	10	34	10	17	04	2C	32	64	39	CC	00	88	9E	04	:9D
5990	E7	0C	EC	62	9E	04	E7	0D	CC	00	88	36	06	EC	62	34	:82
59A0	06	37	10	34	10	17	04	0C	32	64	39	CC	00	28	DD	2A	:E9
59B0	CC	00	88	36	06	CC	2A	16	00	68	CC	00	30	CC	2A	DD	:F3
59C0	2C	DC	2A	C3	00	10	36	06	CC	2C	16	00	48	CC	00	04	:77
59D0	CC	00	00	36	06	CC	2C	36	06	CC	02	7F	36	06	CC	2C	:DD
59E0	36	06	CC	00	01	36	06	CC	00	00	36	06	CC	00	00	34	:4D

59F0	06	37	10	34	10	37	10	34	10	37	10	34	10	37	10	34	:22
Sum:	DB	3F	AF	2C	1												

## リスト 3

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5F00	34	83	00	82	10	26	00	03	17	05	9F	DC	34	83	00	83	:43
5F10	10	26	00	03	17	06	7D	DC	34	83	00	84	10	26	00	03	:23
5F20	17	06	F9	DC	34	83	00	85	10	26	00	03	17	07	FA	DC	:5B
5F30	34	83	00	86	10	26	00	03	17	08	74	DC	34	83	00	87	:23
5F40	10	26	00	03	17	08	EF	DC	34	83	00	88	10	26	00	03	:9B
5F50	17	09	D1	39	DC	34	83	00	03	10	2C	00	10	2C	00	03	:E2
5F60	00	04	DD	42	DC	30	36	06	DC	42	16	00	4B	CC	00	04	:8A
5F70	DC	2E	83	00	86	10	26	00	03	17	08	74	DC	34	83	00	:FF
5F80	36	06	DC	42	36	06	DC	0E	36	06	CC	00	02	36	06	CC	:92
5F90	00	00	34	06	37	10	34	10	37	10	34	10	37	10	34	10	:DB
5FA0	37	10	34	10	37	10	34	10	37	10	34	10	37	10	34	10	:1E
5FB0	6E	DC	42	C3	00	04	DD	42	A3	4A	2F	B1	33	42	DC	32	:3C
5FC0	83	00	0D	10	2F	00	61	DC	26	C3	00	04	DD	42	DC	30	:24
5FD0	36	06	DC	42	36	06	DC	0E	36	06	CC	00	02	36	06	CC	:4D
5FE0	06	DC	42	36	06	DC	2E	C3	00	07	36	06	DC	42	36	06	:CA
5FF0	DC	0E	36	06	CC	00	02	36	06	CC	00	00	34	06	37	10	:7D

Sum: 0B 75 11 0E FD 7B 28 36 39 3D D0 AC 6C E5 AE 36 :99

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6000	34	10	37	10	34	10	37	10	34	10	37	10	34	10	37	10	:2C
6010	34	10	37	10	34	10	17	0E	BB	32	6E	DC	42	C3	FF	FC	:2B
6020	DD	42	A3	C4	2C	B1	33	42	DC	34	83	00	02	10	2C	00	:A9
6030	54	DC	2E	83	00	05	36	06	DC	30	83	00	02	36	06	DC	:CB
6040	2E	C3	00	04	36	06	DC	30	C3	00	02	36	06	DC	06	36	:56
6050	06	CC	00	08	36	06	DC	0E	36	06	CC	00	02	36	06	37	:7B
6060	10	34	10	37	10	34	10	37	10	34	10	37	10	34	10	37	:2C
6070	10	34	10	37	10	34	10	37	10	34	10	17	0D	D8	32	E9	:81
6080	00	16	04	DC	2E	C3	00	05	36	06	DC	30	83	00	00	00	:D7
6090	02	36	06	DC	2E	C3	00	04	36	06	CC	00	C3	00	02	36	:52
60A0	06	DC	06	C3	00	08	36	06	CC	00	08	36	06	DC	0E	36	:1F
60B0	06	CC	00	02	34	06	37	10	34	10	37	10	34	10	17	0D	:6B
60C0	34	10	37	10	34	10	37	10	34	10	37	10	34	10	37	10	:2C
60D0	34	10	17	0D	B1	32	E9	00	10	DC	32	83	00	08	10	2C	:E9
60E0	00	65	DC	2E	C3	00	04	36	06	CC	00	83	00	11	36	06	:4E
60F0	DC	2E	C3	00	0D	36	06	DC	30	83	00	02	36	06	DC	06	:C5

Sum: 3F D6 6E CD 5B 6F 54 D1 70 7A 83 04 E2 80 D9 39 :24

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6100	C3	00	10	36	06	DC	34	36	06	CC	00	20	BD	4D	1F	E3	:5F
6110	C1	36	06	CC	00	16	36	06	DC	0E	36	06	CC	00	02	34	:43
6120	06	37	10	34	10	37	10	34	10	37	10	34	10	37	10	34	:22
6130	10	37	10	34	10	37	10	34	10	37	10	34	10	17	0D	16	:EB
6140	32	E9	00	10	16	00	65	DC	2E	83	00	05	36	06	DC	30	:80
6150	C3	00	02	36	06	DC	2E	C3	00	04	36	06	CC	00	C3	00	:DD
6160	11	36	06	DC	C3	00	10	36	06	DC	34	36	06	CC	00	00	:56
6170	2C	BD	4D	1F	E3	C1	C3	00	16	36	06	CC	00	16	36	06	:2C
6180	DC	0E	36	06	CC	00	02	34	06	37	10	34	10	37	10	34	:34
6190	10	37	10	34	10	37	10	34	10	37	10	34	10	37	10	34	:2C
61A0	10	37	10	34	10	17	0C	AE	32	E9	00	10	39	DC	2E	C3	:9D
61B0	00	0A	36	06	DC	30	C3	00	01	36	06	DC	2E	C3	00	00	:2B
61C0	36	06	CC	00	C3	00	02	36	06	DC	0E	36	06	CC	00	02	:3D
61D0	36	06	CC	00	00	34	06	37	10	34	10	37	10	34	10	37	:BF
61E0	10	34	10	37	10	34	10	37	10	34	10	37	10	34	10	17	:0C
61F0	0C	E2	32	6E	DC	34	83	00	20	DD	34	17	FD	56	39	DC	:D1

Sum: 50 2B 01 F4 A2 DA 5C 0D 0B B9 F6 B4 9B 84 B6 FA :5F

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6200	2E	C3	00	0A	36	06	DC	30	C3	00	01	36	06	DC	2E	C3	:10
6210	00	0D	36	06	DC	30	C3	00	05	36	06	DC	06	C3	02	C2	:C2
6220	36	06	CC	00	06	36	06	DC	0E	36	06	CC	00	02	34	06	:78
6230	37	10	34	10	37	10	34	10	37	10	34	10	37	10	34	10	:2C
6240	37	10	34	10	37	10	34	10	37	10	34	10	17	0C	07	32	:FD
6250	E9	00	10	DC	34	83	00	43	10	26	00	05	CC	00	01	DD	:84
6260	34	DC	34	83	00	41	10	26	00	05	CC	00	02	DD	34	DC	:FE
6270	34	83	00	A5	10	26	00	05	CC	00	03	DD	34	DC	34	83	:0A
6280	00	A3	10	26	00	05	CC	00	04	DD	34	DC	34	83	00	A6	:F8
6290	10	26	00	05	CC	00	05	DD	34	17	FC	B8	39	DC	2E	83	:AE
62A0	00	05	36	06	DC	26	C3	00	0A	36	06	DC	2E	C3	00	04	:1D
62B0	36	06	DC	26	C3	00	19	36	06	DC	06	C3	01	18	36	06	:50
62C0	DC	34	83	00	06	36	06	CC	00	16	BD	4D	1F	E3	C1	36	:8A
62D0	06	CC	00	16	36	06	DC	0E	36	06	CC	00	02	34	06	37	:89
62E0	10	34	10	37	10	34	10	37	10	34	10	37	10	34	10	37	:2C
62F0	10	34	10	37	10	34	10	37	10	34	10	17	0B	58	32	E9	:FF

Sum: 6B 91 73 0F 91 45 CC F5 BE 41 29 AE 34 53 75 C9 :80

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6300	00	10	39	DC	2E	83	00	05	36	06	DC	30	83	00	07	36	:E3
6310	06	DC	2E	C3	00	04	36	06	CC	00	C3	00	08	36	06	DC	:02
6320	06	C3	01	9C	36	06	DC	34	83	00	0C	36	06	CC	00	16	:5F
6330	BD	4D	1F	E3	C1	36	06	CC	00	16	36	06	DC	0E	36	06	:4D
6340	CC	00	02	34	06	37	10	34	10	37	10	34	10	37	10	34	:99
6350	10	37	10	34	10	37	10	34	10	37	10	34	10	37	10	34	:2C
6360	10	17	0A	F2	32	E9	00	10	39	DC	2E	83	00	05	36	06	:55
6370	DC	26	C3	00	0A	36	06	DC	2E	C3	00	04	36	06	DC	26	:1A
6380	C3	00	19	36	06	DC	06	C3	01	DE	36	06	DC	34	83	00	:6B
6390	0F	36	06	CC	00	16	BD	4D	1F	E3	C1	36	06	CC	00	16	:18
63A0	36	06	DC	0E	36	06	CC	00	02	34	06	37	10	34	10	37	:2C
63B0	10	34	10	37	10	34	10	37	10	34	10	37	10	34	10	37	:2C
63C0	10	34	10	37	10	34	10	17	0A	BC	32	E9	00	10	39	DC	:CC
63D0	32	83	00	08	10	2C	00	08	00	00	DD	42	16	00	05	07	
63E0	CC	00	01	DD	42	DC	2E	83	00	05	36	06	DC	30	83	00	:49
63F0	0F	36	06	DC	2E	C3	00	04	36	06	CC	00	C3	00	08	36	:5D

Sum: BE CD 88 B7 53 7B 1B 4C 5A 19 80 01 A6 47 DC 5D :19

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6400	06	DC	06	C3	02	96	36	06	DC	42	36	06	CC	00	16	BD	:78
6410	4D	1F	E3	C1	36	06	CC	00	16	36	06	DC	0E	36	06	CC	:5C

6420	00	02	34	06	37	10	34	10	37	10	34	10	37	10	34	10	:DD
6430	37	10	34	10	37	10	34	10	37	10	34	10	37	10	34	10	:2C
6440	17	0A	13	32	E9	00	1D	39	CC	00	0A	36	04	DC	26	03	:F8
6450	00	04	36	06	CC	00	1D	36	06	DC	26	03	00	1F	36	06	:85
6460	DC	06	C3	02	0A	36	06	DC	34	83	00	80	36	06	CC	00	:08
6470	46	BD	4D	1F	E3	C1	36	06	CC	00	46	36	06	DC	0E	36	:8D
6480	06	CC	00	02	34	06	37	10	34	10	37	10	34	10	37	10	:6B
6490	34	10	37	10	34	10	37	10	34	10	37	10	34	10	37	10	:2C
64A0	34	10	17	09	B1	32	E9	00	1D	39	CC	32	DC	42	CC	00	:72
64B0	00	DD	44	CC	00	00	DD	34	CC	00	2B	DD	2E	DC	3B	83	:A0
64C0	00	01	9E	04	58	49	EC	8B	1F	89	4F	83	00	80	10	26	:E8
64D0	00	08	CC	00	00	DD	32	16	00	05	CC	00	0A	DD	32	DC	:CB
64E0	42	83	00	00	10	2F	00	AB	DC	2B	36	06	DC	32	5B	49	:9E
64F0	A3	C1	53	43	C3	00	01	DD	30	17	FE	07	DC	2E	C3	00	:B4

6960	24	DC	28	36	06	DC	24	16	00	4B	CC	00	04	DC	46	83	:3A
6970	00	0B	36	06	DC	24	36	06	DC	46	C3	00	07	36	06	DC	:84
6980	24	36	06	DC	0E	36	06	CC	00	04	36	06	CC	00	00	34	:92
6990	06	37	10	34	10	37	10	34	10	37	10	34	10	37	10	34	:22
69A0	10	37	10	34	10	37	10	34	10	37	10	34	10	37	10	34	:0A
69B0	C3	00	04	DD	24	A3	C4	2F	B1	33	42	39	DC	18	36	06	:ED
69C0	CC	00	19	BD	40	1F	C3	00	87	DD	46	DC	28	36	06	DC	:97
69D0	1E	58	49	A3	C1	53	43	C3	00	01	DD	24	DC	46	83	00	:23
69E0	05	36	06	DC	24	B3	00	01	36	06	CC	00	05	36	06	CC	:00
69F0	06	DC	24	C3	00	01	36	06	CC	00	05	36	06	CC	00	04	:E3

Sum: CF 9B BB B0 A4 7D BB C1 36 91 F8 AF 0F E7 03 6E :3E

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6A00	36	06	CC	00	02	34	06	37	10	34	10	37	10	34	10	37	:91
6A10	10	34	10	37	10	34	10	37	10	34	10	37	10	34	10	37	:0C
6A20	04	B2	32	4E	39	DC	1A	B3	00	13	10	26	09	CC	00	00	:26
6A30	07	BD	4E	DE	16	00	63	DC	1A	DD	40	DC	18	36	06	DC	:8B
6A40	40	16	00	21	CC	FF	FF	DC	40	C3	00	01	58	49	9E	1C	:7C
6A50	30	B8	36	10	DD	0C	4E	1C	58	49	EC	BB	ED	D1	DC	40	:C9
6A60	C3	FF	DD	40	A3	C4	2F	1C	DB	33	42	DC	1A	C3	00	01	:7B
6A70	DD	1A	DC	18	58	49	9E	1C	30	B8	36	10	DD	20	36	06	:7F
6A80	CC	01	00	BD	40	1F	D3	1E	ED	D1	DC	18	34	06	17	F2	:DC
6A90	33	62	DC	1B	C3	00	01	DD	18	39	DC	18	93	1A	10	A1	:5E
6AA0	26	00	09	CC	00	07	BD	4E	16	00	41	DC	1A	B3	00	:BB	
6AB0	01	DD	1A	DC	18	DD	40	DC	1A	36	06	DC	40	16	00	1E	:BB
6AC0	DC	40	58	49	9E	1C	30	B8	36	10	DD	40	C3	00	01	9E	:F6
6AD0	1C	58	49	EC	BB	ED	D1	DC	40	C3	00	01	DD	40	A3	C4	:56
6AE0	2F	DE	33	42	DC	1B	34	06	17	F1	D9	32	62	39	9E	00	:FC
6AF0	EC	B4	B3	00	00	10	27	00	0E	DC	12	34	06	17	00	3F	:B6

Sum: 9A 6D 49 61 23 66 BE C3 3A F7 B6 A0 E3 FD 98 4E :0B

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6B00	32	62	DD	22	16	00	37	DC	08	DD	46	DC	46	93	0A	10	:B6
6B10	2C	00	18	CC	00	00	9E	46	ED	B4	CC	00	00	9E	46	ED	:02
6B20	02	DC	46	C3	00	00	4D	46	16	FF	0E	CC	FF	FF	9E	08	:AF
6B30	ED	B4	CC	FF	FF	DD	22	CC	00	01	9E	00	ED	B4	39	DC	:28
6B40	08	DD	46	CC	00	00	DD	24	DC	46	36	06	CC	FF	FF	BD	:E4
6B50	4D	C1	36	06	DC	24	36	06	EC	62	BD	4D	A1	A4	C0	E4	:C7
6B60	00	10	27	00	10	9E	46	EC	B4	DD	46	DC	24	C3	00	01	:42
6B70	DD	24	16	FF	DC	46	DD	36	39	DC	08	DD	46	9E	46	:42	
6B80	EC	B4	36	06	CC	00	00	BD	40	C1	36	06	9E	46	EC	:51	
6B90	36	06	CC	00	00	BD	40	C1	AA	C0	EA	C0	36	06	DC	:46	
6BA0	36	06	DC	0A	BD	40	A1	A4	C0	E4	C0	10	27	00	0A	:F2	
6BB0	46	C3	00	40	DD	46	16	FF	C5	DC	46	DD	36	39	DC	:B2	
6BC0	B3	FF	FF	10	26	00	06	17	01	CC	16	00	26	DC	12	:83	
6BD0	00	01	34	06	17	FF	68	32	62	DD	48	9E	22	EC	B4	:9E	
6BE0	48	ED	B4	CC	00	00	9E	22	ED	B4	17	00	31	9E	48	:D0	
6BF0	B4	DD	22	39	DC	12	B3	00	01	34	06	17	FF	41	32	:53	

Sum: 2C B1 77 EC 53 1C 06 B3 5A C1 46 47 49 BC 42 7E :A5

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6C00	DD	48	17	FF	75	DD	4A	DC	4A	9E	48	ED	B4	CC	22	9E	:F0
6C10	4A	ED	B4	CC	00	DD	22	17	00	C1	17	01	D9	39	DC	22	:E0
6C20	DD	4C	CC	00	00	DD	42	CC	00	00	DD	44	CC	00	02	DD	:AC
6C30	4E	DC	42	83	00	04	10	2C	00	99	DC	4A	36	06	DC	:02	
6C40	36	06	CC	00	00	BD	40	1F	E3	C1	DD	50	DC	44	58	49	:23
6C50	9E	50	30	B8	36	10	DD	4E	9E	4C	58	49	EC	BB	ED	D1	:D9
6C60	DC	44	C3	00	01	DD	44	DC	4E	C3	00	01	DD	4E	DC	44	:9E
6C70	B3	00	01	9E	50	58	49	EC	BB	36	06	CC	FF	FF	BD	40	:43A
6C80	B9	36	06	DC	4E	36	06	CC	00	20	BD	40	B9	AA	C0	EA	:5E
6C90	C0	27	B9	DC	44	83	00	01	9E	50	58	49	EC	BB	83	FF	:CC
6CA0	FF	10	26	00	0C	DC	42	C3	00	01	DD	42	CC	00	00	DD	:EB
6CB0	44	DC	4E	83	00	20	10	26	00	16	9E	4C	ED	CC	02	52	:64
6CC0	CC	00	00	9E	4C	ED	02	DC	52	DD	4C	CC	00	02	DD	4C	:F5
6CD0	16	FF	5E	CC	00	00	9E	4C	ED	02	39	DC	22	DD	4C	44	:44
6CE0	00	00	DD	42	CC	00	00	DD	44	CC	00	02	DD	4E	DC	42	:23
6CF0	B3	00	04	10	2C	00	9E	DC	4A	36	06	DC	42	36	06	CC	:A3

Sum: A6 3F DB 7E 8B 3F 0A B7 C9 66 6E 46 A1 D1 E5 CA :CA

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6D00	00	60	BD	40	1F	E3	C1	DD	50	DC	4E	58	49	9E	4C	30	:3F
6D10	B8	36	10	DD	4A	9E	50	58	49	EC	BB	ED	D1	DC	4E	C3	:A2
6D20	00	01	DD	4E	DC	44	C3	00	01	DD	44	BD	44	83	00	01	:D5
6D30	9E	50	58	49	EC	BB	36	06	CC	01	FF	FF	BD	40	B9	36	:06
6D40	DC	4E	36	06	CC	00	20	BD	40	B9	AA	C0	EA	C0	27	B9	:09
6D50	DC	44	B3	00	01	9E	50	58	49	EC	BB	83	FF	FF	10	26	:61
6D60	00	0C	DC	42	C3	00	01	DD	42	CC	00	00	DD	44	DC	:4E	:24
6D70	B3	00	20	10	26	00	1B	17	FE	00	DD	52	DC	52	9E	:4C	:50
6D80	ED	02	DC	52	DD	4C	CC	FF	FF	9E	4C	ED	02	CC	00	02	:B7
6D90	DD	4E	16	FF	59	39	CC	00	00	9E	02	ED	08	CC	00	00	:FF
6DA0	9E	02	ED	0A	CC	00	00	9E	02	ED	0E	CC	00	00	DD	5A	:F8
6DB0	CC	00	09	36	06	DC	54	16	00	1B	DC	54	58	49	9E	04	:E5
6DC0	30	B8	36	10	DD	4A	9E	02	58	49	EC	BB	ED	D1	DC	:54	:D7
6DD0	C3	00	01	DD	54	A3	C4	2F	E1	33	42	CC	FF	FF	9E	:04	:4D
6DE0	ED	B8	60	CC	FF	FF	9E	04	ED	B9	00	C0	CC	FF	FF	9E	:D3
6DF0	04	ED	B9	01	20	39	CC	00	00	9E	04	ED	08	CC	00	00	:03

Sum: 7C D7 BF 63 38 7E 4E 2C 63 FC 9B 71 6F 87 75 C3 :3B

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
6E00	9E	04	ED	0A	CC	00	00	9E	04	ED	0E	CC	00	00	DD	5A	:FF	
6E10	CC	00	09	36	06	DC	54	16	00	1B	DC	54	58	49	9E	:02	:E3	
6E20	30	B8	36	10	DD	4A	9E	04	58	49	EC	BB	ED	D1	DC	:54	:D9	
6E30	C3	00	01	DD	54	A3	C4	2F	E1	33	42	CC	FF	FF	9E	:04	:FD	
6E40	ED	B8	60	CC	FF	FF	9E	04	ED	B9	00	C0	CC	FF	FF	9E	:04	:D0
6E50	04	ED	B9	01	20	39	CC	00	00	9E	04	ED	08	CC	00	00	:03	:D8
6E60	CC	00	09	36	06	DC	58	9E	58	CC	00	1C	E7	80	EC	:62	:E1	:80
6E70	EC	64	ED	81	EC	66	ED	81	EC	66	ED	81	EC	66	E7	:80	:01	:A1
6E80	EC	E8	10	E7	80	EC	6C	E7	80	9F	58	EC	6A	DD	5A	:C0	:5A	:5A

6E90	00	00	DD	5C	DC	56	36	06	DC	5C	16	00	18	DC	5C	9E	:E3
6EA0	58	30	B8	36	10	DD	4C	5C	9E	5A	E6	B8	4F	E7	D1	DC	:39
6EB0	C3	00	01	DD	5C	A3	C4	2F	E4	33	42	17	00	B8	39	:93	
6EC0	00	00	9E	02	ED	08	CC	00	00	9E	02	ED	0A	CC	00	:54	
6ED0	9E	02	ED	0E	17	00	61	CC	FC	B2	4D	58	9E	58	CC	:00	
6EE0	15	E7	80	EC	6A	E7	80	EC	6C	E7	80	EC	62	ED	81	:E0	
6EF0	64	ED	81	EC	66	ED	81	EC	68	ED	81	EC	6E	E7	80	:F0	
Sum:	24	52	BA	96	01	AC	75	02	03	99	0B	B4	1F	0C	60	:4F	

## リスト 3

73C0	72	6D	3F	00	BD	4D	E2	DD	2C	BD	4D	17	DC	2C	36	06	:78
73D0	CC	00	00	BD	4D	99	36	06	CC	00	0F	BD	:E7				
73E0	4D	B1	A4	C0	E4	C0	27	CB	BD	4D	17	BD	4D	0B	45	6E	:E1
73F0	76	65	6C	6F	70	65	20	54	69	6D	65	28	78	30	2E	30	:68
Sum:	0C	D6	AF	EB	85	C2	46	3A	FA	41	91	85	DF	94	52	41	:97
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
7400	31	73	29	3F	61	62	6C	6D	E2	DD	2E	BD	4D	17	CC	00	:20
7410	36	06	DC	2E	BD	4D	1F	DD	30	39	BD	4D	0B	4E	6F	69	:F0
7420	73	65	20	43	6F	6E	74	72	6F	6C	00	BD	4D	17	BD	4D	:04
7430	17	DC	08	34	06	17	00	57	32	62	BD	4D	17	BD	4D	0B	:6D
7440	4E	6F	69	73	65	20	46	72	65	71	75	65	6E	63	79	28	:F8
7450	78	31	30	30	48	7A	29	3F	00	BD	4D	E2	DD	2E	BD	4D	:34
7460	17	CC	03	00	36	06	CC	2E	BD	4D	38	DD	32	39	BD	4D	:C3
7470	0B	4C	65	76	65	6C	20	43	6F	6E	74	72	6F	6C	00	BD	:C1
7480	4D	17	BD	4D	17	DC	0A	34	06	17	00	00	32	62	39	EC	:45
7490	62	DD	34	CC	00	00	DD	36	CC	00	02	36	06	DC	36	16	:84
74A0	00	B1	BD	4D	0B	50	53	47	2D	00	CC	00	41	D3	36	BD	:80
74B0	4E	DE	BD	4D	0B	20	20	00	DC	36	9E	34	E6	BB	4F	63	:A8
74C0	00	00	10	26	00	0E	BD	4D	0B	64	69	73	61	62	6C	65	:2D
74D0	00	16	00	0B	BD	4D	0B	65	6E	61	62	6C	65	20	00	BD	:7A
74E0	4D	0B	20	20	3D	20	20	00	BD	4D	62	BD	38	DC	38	83	:A1
74F0	00	00	27	F4	DC	38	83	00	44	10	26	00	0B	BD	4D	0B	:4C
Sum:	23	16	F0	F5	7D	9A	10	0D	94	BD	DB	30	DA	DB	51	62	:E6
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
7500	64	69	73	61	62	6C	6D	65	00	CC	3B	83	00	45	10	26	:00
7510	0A	BD	4D	0B	65	6E	61	62	6C	65	00	BD	4D	17	DC	36	:B9
7520	9E	34	30	BB	36	10	DC	3B	83	00	44	4F	E7	D1	DC	38	:C9
7530	36	06	CC	00	44	BD	4D	0B	36	06	CC	00	00	00	00	00	:67
7540	45	BD	4D	0B	9A	CC	0A	E4	00	10	27	FF	56	DC	36	C3	:00
7550	01	DD	36	A3	C4	10	2F	FF	49	33	42	39	EC	62	DD	3A	:15
7560	CC	00	00	00	DD	36	CC	00	02	36	06	CC	36	16	00	66	:34
7570	4D	0B	50	53	47	2D	00	CC	00	41	D3	36	BD	4E	DE	BD	:2B
7580	4D	0B	20	20	3D	20	20	00	CC	00	4A	36	06	CC	00	00	:F8
7590	4F	BD	4E	8A	BD	4D	0B	20	20	3D	20	20	00	BD	4D	E2	:A2
75A0	DD	3C	BD	4D	17	DC	36	9E	34	30	BB	36	10	DC	3C	4F	:BC
75B0	E7	D1	DC	3C	36	06	CC	00	00	BD	4D	99	36	06	CC	00	:CF
75C0	06	6C	CC	00	0F	BD	4D	B1	A4	C0	E4	C0	27	A1	DC	36	:B4
75D0	C3	00	01	DD	36	A3	C4	10	2F	96	33	42	39	DC	20	9E	:04
75E0	E6	BB	4F	83	00	01	10	26	00	0A	CC	00	50	34	06	17	:F1
75F0	06	C1	32	62	DC	20	9E	0B	E6	88	4F	83	00	01	10	26	:77
Sum:	E6	2C	E4	78	57	EC	D4	B0	40	FC	AB	E0	B1	B3	69	91	:27
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
7600	00	1C	CC	00	52	34	06	17	06	A9	32	62	CC	00	62	36	:32
7610	06	DC	32	34	06	17	06	10	34	10	17	06	84	32	64	DC	:3C
7620	9E	06	E6	88	4F	83	00	01	10	26	00	2B	CC	00	55	34	:9E
7630	06	17	06	7F	32	62	CC	00	40	D3	2C	34	06	17	06	73	:0B
7640	32	62	CC	00	63	6C	CC	00	30	34	06	37	10	34	10	17	:E7
7650	06	7E	32	64	16	00	1F	CC	00	54	34	06	17	06	54	32	:4C
7660	62	CC	00	30	36	06	CC	00	20	9E	0A	E6	88	4F	E3	C1	:4D
7670	06	17	06	3F	32	62	39	EC	0E	02	4F	E6	0B	83	00	87	:10
7680	26	00	AE	DC	1E	C3	00	01	DD	3E	9E	02	EC	88	10	DD	:AE
7690	40	CC	27	10	DD	42	DC	40	36	06	CC	DC	4D	BD	4D	38	:A0
76A0	00	00	10	27	00	20	CC	00	70	36	06	CC	DC	40	36	06	:03
76B0	42	BD	4D	3B	E3	C1	36	06	CC	00	00	34	06	37	10	34	:E9
76C0	10	17	06	0C	32	64	DC	40	36	06	CC	DC	42	34	06	37	:10
76D0	34	10	17	05	0C	32	64	DC	40	36	06	CC	DC	42	34	06	:10
76E0	BD	4D	3B	DD	42	DC	42	83	00	00	26	AA	DC	1E	C3	00	:92
76F0	01	DD	44	CC	00	70	36	06	CC	00	00	34	06	37	10	34	:1B
Sum:	F4	B2	BC	19	CC	B6	B2	9F	C7	F6	2E	EF	D4	01	B0	48	:F5
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
7700	10	17	05	CC	32	64	DC	4E	7E	93	44	10	2C	00	12	DC	:4E
7710	93	3E	3E	ED	94	4C	83	00	02	9E	44	ED	84	16	00	00	:EB
7720	17	00	12	CC	1E	93	44	83	00	02	9E	44	ED	84	16	00	:EB
7730	03	17	00	01	39	CC	00	00	DD	36	CC	00	06	36	06	CC	:1D
7740	36	16	00	14	DC	36	9E	0E	30	BB	36	10	CC	00	00	E7	:D2
7750	D1	CC	36	C3	00	01	DD	36	A3	C4	2F	EB	33	42	DC	20	:A9
7760	9E	02	58	49	EC	BB	1F	89	4F	83	00	80	10	26	00	0B	:F0
7770	CC	00	13	DD	46	16	00	05	CC	00	07	DD	46	9E	02	4F	:0F
7780	E4	06	83	00	82	10	26	00	46	CC	00	03	DD	36	9E	02	:EF
7790	4F	E4	07	DD	48	DC	48	83	00	00	10	2F	00	2E	DC	36	:87
77A0	36	06	CC	00	07	34	06	37	10	34	10	17	04	E7	32	64	:6C
77B0	9E	0E	30	BB	36	10	CC	00	01	E7	D1	DC	36	C3	00	04	:0B
77C0	DD	36	CC	48	83	00	01	DD	48	16	FF	C9	16	00	43	CC	:E3
77D0	00	06	DD	36	9E	02	4F	E6	07	DD	48	DC	48	83	00	00	:C1
77E0	10	2F	00	2E	DC	36	9E	02	4F	E6	07	DD	48	83	00	00	:C1
77F0	10	17	04	A1	32	64	9E	0E	30	BB	36	10	CC	FF	FF	E7	:C0
Sum:	34	E2	99	99	BA	EB	FA	62	C3	B3	5E	B0	CD	AF	D3	EB	:07
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
7800	D1	CC	36	C3	00	03	DD	36	CC	00	01	DD	36	C3	00	16	:9F
7810	FF	C9	9E	02	4F	E6	0A	83	00	84	10	26	00	14	CC	00	:C4
7820	80	34	06	17	04	BD	32	62	CC	03	E7	DD	24	CC	03	E7	:63
7830	DD	24	9E	02	4F	E6	0D	93	24	10	27	00	1C	CC	00	60	:1B
7840	36	06	9E	02	4F	E6	0D	34	06	37	10	34	10	17	04	80	:7E
7850	32	64	9E	02	4F	E6	0D	24	17	00	36	9E	02	4F	E6	0D	:9B
7860	0E	83	00	85	10	26	00	0A	CC	00	81	34	06	17	04	43	:78
7870	32	62	9E	02	4F	E6	0E	83	00	86	10	26	00	13	CC	00	:95
7880	2C	36	06	CC	00	00	34	06	37	10	34	10	17	04	41	32	:87
7890	64	39	CC	00	00	DD	4A	CC	00	00	DD	4C	CC	00	00	DD	:2E
78A0	4E	DC	4E	9E	22	58	49	EC	8B	83	FF	FF	10	27	01	43	:4C
78B0	DC	4E	9E	22	58	49	EC	8B	1F	89	4F	DD	50	DC	4E	9E	:EE
78C0	22	58	49	EC	8B	4F	DD	52	DC	50	36	06	CC	00	00	BD	:A9
78D0	4D	99	36	06	CC	50	36	06	CC	00	05	BD	4D	B1	A4	C0	:7A
78E0	E4	C0	10	27	00	09	DC	50	58	49	DD	54	17	01	05	DC	:DB

78F0	50	36	06	CC	00	21	BD	4D	99	36	06	DC	50	36	06	CC	:8C
Sum:	32	CE	A5	DA	B0	7B	AD	8A	3C	9E	BF	F2	B8	BB	79	1B	:43
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
7900	00																





# FM-7/FM-8

## F-BASIC

〔BASICとモニタの  
コマンドを拡張!〕

# EXPANDER

エキスパンダ

ETF Laboratory 笠作貴弥

私はFM-8を手に入れて約1年以上になりますが、いろいろと使っているうちに、次のような不満がでてきました。

- ①モニタ機能が貧弱。
- ②音楽機能がない。
- ③グラフィック画面のウィンドウ・コピーができない。
- ④漢字フォントを使いにくい（プリンタ出力ができない）。

特に①は、主にマシン語プログラムを作っている私には、決定的なことです。

そこで、以上の不満を解決すべく今回のプログラムを制作しました。

また、FM-7を入手したので移植しました。

## プログラムの概要

F-BASICでは、DISK BASIC用に\$E8～\$FEまでの中間コードが割り当てられており、これらの中間コードは、ROM版BASICでは使っていません。そこで、これらの中間コードに以下に挙げるようなステートメントを定義し、F-BASICをより使い易くしようというものです。ですから、このプログラムは、DISKシステムでは使うことができません。しかし、拡張モニタ部分はDISKシステムでも使用可能です。

## 追加・強化されるステートメント

★COPY	ウィンドウ・コピー	} PSGカード用
★KSYMBOL	漢字シンボル文	
★KPRINT	漢字フォントのプリンタ出力	
●SCROLL	グラフィック画面のスクロール	
●CRT	画面表示を禁止/解除	
●PSG	PSGヘデータを転送	
●MUSIC	音楽演奏	
●TEMPO	MUSIC文のテンポを設定	
★MON	コマンドを増強	

●印はFM-8のみ

## プログラムの入力・使い方

モニタ、または何らかのマシン語モニタを使って入力してください。

入力し終わったら、FM-8ではSAVEM"EXPANDER",&H6800,&H7DFF,&H6800としてセーブしておきましょう。次にEXEC&H6800を実行すると、

Expanded MONITOR!!

Expanded F-BASIC!!

と表示され、拡張命令を使うことができます。次回からはFM-8をスイッチON後、LOADM"EXPANDER",,Rとして、このプログラムをロードすればOKです。CLEAR文の設定も自動的に行なわれます。

FM-7では、プログラムは256バイト単位でのリロケータブルになっているので、\$△△00から入力して使えます。DISKシステムではそのまま、通常のBASICでは+\$1000番地から入力すると良いでしょう。

入力し終わったらSAVEM"EX-BASIC",START,END,EXEC RETURNとしてセーブしておきましょう。

次にEXECSTART RETURN とすると"Expanded F BASIC"と表示され、拡張命令を使うことができます。次回からは、LOADM "EX BASIC",,R RETURNとしてこのプログラムをロードすればOKです。EX-MONも同様です(START,END,EXECは入力アドレス)。

## マニュアル

### ★COPY .....

書式: COPY (X<sub>1</sub>, Y<sub>1</sub>)-(X<sub>2</sub>, Y<sub>2</sub>) [, オフセット]

機能: グラフィック画面の指定されたエリアのみプリンタへハード・コピーします。

解説: 座標指定は、180×25のキャラクタ座標で指定します。オフセットはプリンタ用紙のどの位置にコピーするか指定するもので、用紙の左から何ドット分オフセットするかを指定します。省略するとオフセットなしと見なします。

COPY(0,0)-(79,24),160 とすれば、HARDC2と同じです。COPY命令でコピーした方が40秒ぐらい速くなります。

この命令は、初めドット座標でウィンドウを設定するようにしたのですが、プログラムがかなり長くなったので、このようにしました。

### ★KSYMBOL .....

書式: KSYMBOL(X,Y), 横倍率, 縦倍率[, [カラーコード][機能]] ; 漢字コード [, 漢字コード] ..

機能: 漢字フォントを画面上の任意の場所に任意の大きさで表示します。

解説: この部分は、長山泰光氏が'82年6月号で発表されたものをそのまま使わせて頂きました。

勝手に使ってしまったことを誌面をお借りして、お詫びいたします。

なお、このプログラム全体を作るにあたって、長山氏のプログラムをかなり参考にさせて頂きました。

COPY文とこのKSYMBOL文を組み合わせれば、かなり応用が効くと思います。

## ★KPRINT .....

書式: KPRINT 漢字コード [, 漢字コード] ..... [, /]

機能: 漢字フォントを直接プリンタ (MB27401またはMP-80TYPEII) に転送し印字する。

解説: バッファフル(48文字)または、漢字コードの代わりに“/”で印字改行します。

印字はプリンタの標準外の機能コマンドESC“J”を使って、圧縮印字するので通常文字とほぼ同じ大きさで印字されます。

FM-7では以下の機能キャラクタもあります。

#...バッファ・ワーク・エリアのイニシャライズ

[...倍文字指定

]...倍文字モード解除

@...たて2倍文字指定

以上の機能キャラクタを漢字コードの代わりに使うことにより、4種類の文字の大きさを選択できます。ただし通常文字と縦2倍文字は同一行に混在できません。

## SCROLL .....

書式: SCROLL オフセット値, 方向

機能: VRAMにオフセットをかけ、擬似的にスクロールさせます。

解説: オフセット値160で、1ドット行スクロールします。方向は0で通常、1で逆方向のスクロールを行いません。

この命令はBIOSのメンテナンス・コマンドを使って、サブシステムのハードを直接コントロールしているので非常に高速です。ゲームなどに応用できると思います。

## CRT .....

書式: CRT ON/OFF

機能: 画面の表示を有効にしたり、禁止したりします。

解説: 画面を点滅させたり、CRTOFFにより表示を禁止しておき、他の命令を使って図形を画面に描き、CRT ONで表示させれば一度に図形が浮び上がります。

## PSG .....

書式: PSG レジスタ番号, データ

機能: PSGへデータを転送します。

## MUSIC .....

書式: MUSIC 文字式

機能: 文字式で示されたデータを音階データと見なし、そ

表1

①オクターブ指定				
コード	L	M	H	
オクターブ	低	中	高	
②音形指定				
コード	P1	P2	P3	P4
長さ	25%	50%	75%	100%

### ③音長指定

コード	0	1	2	3	4	5	6	7	8	9
音 符	♪	♪	♪	♪	♪	♪	♪	♪	♪	♪

### ④音程指定

コード	C	D	E	F	G	A	B	#	R
音 程	ド	レ	ミ	ファ	ソ	ラ	シ	半音上げる	休 符

1つの音符はオクターブ指定、音形指定、音長指定、音程指定によって構成されます。このうち前の音符と等しい場合、音程データを除いて省略することができます。

①オクターブ指定 (L・M・H)

音階レンジの切り換えです。

②音形指定 (P1～P4)

同じ音長でも、スタッカート気味とか、テヌート気味とかの変化をつけます。

③音長指定 (0～9)

音符の長さを指定します。

④音程指定 (CDEFGABC) (#) (R)

音程を指定します。半音は#のみ有効で音程指定の次に#を書きます。またRのときは、休符となります。

れを演奏します。

解説: 音階データの書式は表1を参考にしてください。音形・音長・オクターブ指定は省略することができます。その場合は、それ以前で指定されたのと同じになります。

## TEMPO .....

書式: TEMPO 式

機能: MUSIC文の演奏速度を指定します。式の値は1～255で、大きいほど遅くなります。

解説: PSG、MUSIC、TEMPOステートメントは、当然のことながらPSGカードがないと使えません。基板サービスもされているので、お持ちでない方は早速製作しましょう。

## ★MON .....

このコマンドは『FM-8活用研究』に発表したデバッグを改良、整理して、F-BASICとリンクしたものです。

★Bコマンド

ブレーク・ポイントを設定します。アドレスを入力するとそこへブレーク・ポイントがセットされます。“\*”を入力すると解除されます。

★Dコマンド

メモリ・ダンプのコマンドです。オプションを指定することにより、プリンタへも出力することができます。ダンプの一時停止はESCで、中断はSTOPで行ないます。

また、カーソルを移動して同時に複数行のエディットができます。

★Gコマンド

F-BASICのGコマンドと同様です。

★Iコマンド

指定されたアドレスより、入力された文字列をキャラクタ・コードに直してメモリにストアします。

★Kコマンド

指定されたメモリ・エリアを同一データでクリアします。

★Lコマンド

ディスプレイのコマンドです。オプションの指定により、プリンタへも出力可能です。また、マシン・コード部分のエディットができます。

★Mコマンド

メモリ・チェンジのコマンドです。このコマンド実行中に限りテン・キー部分が16進キーになります。←でバック、↓でインクリメントします。↑を入力すると16バイト(画面が40字モードのときは8バイト)バックし、エディット・モードになり、スクリーン・エディットができます。

また、16バイト入力するごとにベルが鳴り、チェック・

表2 拡張モニタ コマンド一覧

コマンド	書式・機能
B	B RETURN 1**** アドレス RETURN
D	D RETURN D 先頭アドレス RETURN 8行ダンプ D 先頭アドレス, 最終アドレス (, Por, AP) RETURN
G	G RETURN G アドレス RETURN
I	I 先頭アドレス, 文字列 RETURN
K	K 先頭アドレス, 最終アドレス, データ RETURN
L	L RETURN L 先頭アドレス RETURN 15行ディスアSEMBル L 先頭アドレス, 最終アドレス (, P) RETURN
M	M RETURN M 先頭アドレス RETURN
Q	Q RETURN
R	R RETURN F-BASICに同じ
X	X 先頭アドレス, 最終アドレス, 転送先頭アドレス RETURN

サムを出力します。16進以外のキー入力で、コマンド待ちに戻ります (図1)。

#### ★Qコマンド(FM-7のみ)

拡張モニタ切り離し。通常のモニタをリンクしてBASICのコマンド待ちに戻ります。

#### ★Rコマンド

F-BASICのRコマンドと同様です。

#### ★Xコマンド

トランスファのコマンドです。転送エリアが重なっていてもかまいません。

G, I, K, Xコマンドについては処理する前に確認のため, "Are you sure?(Y/N)" と聞いてくるのでYまたは, Nを入力してください。

■各コマンドの書式一覧表を表2に示します。

モニタ・モードより, F-BASICへ戻るには **CTRL** - **C** または **X**, **STOP** を入力します。

## 新ステートメント定義の方法

F-BASICではDISKシステムのために, ワーク・エリアにジャンプ・テーブルがあります。\$1F9~\$1FDの5バイトがそれで, \$1F9に追加ステートメント数, \$1FA, 1FBに追加ステートメントのキーワード・テーブル, \$1FC, 1FDに追加ステートメントの処理先アドレスをセットします。

キーワード・テーブルは, キーワードの終わりの文字のビット7を1にしたものを並べます。テーブルの最後は, \$00で示します。

処理先ではAレジスタに中間コードを入れてジャンプし

て来るので, それによって各ルーチンへジャンプするようにすればOKです。詳しくはリスト1を参照してください。

## その他

FM-8 拡張BASIC部分と拡張モニタを合わせると約6Kバイトもあり, フリーエリアが大幅に減少してしまいます。拡張BASICとモニタ部は, 独立して動作させることができるので, それぞれ別々のテープを作ることをお勧めします。その場合は, 次のようにします。

SAVEM"EX-MON",&H7000,&H7DFF,&H7000

これで, 拡張モニタのみのテープができました。

次に,

SAVEM"EX-BASIC",&H6800,&H6E90,&H6800

LOADM"EX-BASIC",&HF00

\$7700~\$7702を\$12 (NOP) にする。

SAVEM"EX-BASIC",&H7700,&H7D90,&H7700

これで, 拡張BASICのみのテープができます。

図1 テン・キーの16進化

FM-8			
←		F	

7	8	9	E
4	5	6	D
1	2	3	C
0	A	B	

FM-7			
A	B	C	D
7	8	9	E
4	5	6	F
1	2	3	↶
0			

## おわりに

結構便利なプログラムができたと思いますが, いかがでしょうか?

KPRINT文, COPY文などは非常に役立つと思います。みなさんもぜひ使ってください。

#### 参考文献

- 1) "PSGカードの製作": FM-8活用研究
  - 2) "KSYMBOL文": I/O'82年6月号
  - 3) "FM-8のROMを解読": I/O'81年12月号
  - 4) "6809デバッグ": FM-8活用研究
- その他6809, FM-8, L-3に関する記事

#### デモ・プログラム

```

10 * EXPANDED F-BASIC DEMO
20 WIDTH40,25:TEMPO 10
30 FOR X=0 TO 39:COLORX MOD6+1:PRINT"■";:MUSIC"P11CEG":NEXT
40 LOCATE 0,15
50 FOR X=0 TO 39:COLORX MOD6+1:PRINT"■";:MUSIC"P11CEG":NEXT
60 SYMBOL (30,40),"DEMO PROGRAM",6,8,5
70 TEMPO 150:MUSIC "P13LCDEFGABMCDEFGABHCDEFGAB"
80 FOR J=1 TO 10
90   FOR I=1 TO 30:SCROLL 160,1:NEXT
100  FOR I=1 TO 30:SCROLL 160,0:NEXT
110 NEXT
120 FOR J=1 TO 10
130  FOR I=1 TO 100:CRTOFF:NEXT
140  FOR I=1 TO 100:CRTON:NEXT

```

## デモ・プログラム

```

150 NEXT
160 PSG 7,&HF7:PSG 8,16:PSG 11,0:PSG 12,5:PSG 6,1
170 FOR J=1 TO 3
180   FOR I=1 TO 500:NEXT:PSG 13,9
190 NEXT
200 FOR I=&H3021 TO &H307E
210   KPRINT I
220 NEXT:KPRINT /
230 COPY (0,0)-(79,24)
240 END

```

## リスト I (FM-8用)

```

00001          NAM      EXPANDER
00002          *****
00003          *  ♥♥ F-BASIC EXPANDER ♥♥ *
00004          *      Version 1.03      *
00005          * (C) ETF Laboratory    *
00006          *  Designed by T.Kasasaku *
00007          *  HERO-09 ASSEMBLER !!  *
00008          *****
00009          OPT      M,N06
00010          6800     BEGIN    EQU    $6800
00011          0000     SETDP    $00
00012          6800     ORG      BEGIN
00013          6800 17 081A 701D    LBSR    $701D    for MON APPEND
00014          6803 30 8C FB      LEAX    BEGIN-2,PCR
00015          6806 9C 45        CMPX    $45
00016          680B 24 11 681B    BCC     HITOMI
00017          680A DC 45        LDD     $45
00018          680C 93 3F        SUBD    $3F
00019          680E 34 06        PSHS    A,B
00020          6810 9F 45        STX     $45
00021          6812 1F 10        TFR     X,D
00022          6814 A3 E1        SUBD    ,S++
00023          6816 DD 3F        STD     $3F
00024          6818 BD 9414      JSR     $9414    All Variables Clear
00025          681B 86 08        HITOMI LDA    #8      <--
00026          681D B7 01F9      STA     $1F9      <--
00027          6820 30 8D 0051    LEAX    CODE,PCR <--
00028          6824 BF 01FA      STX     $1FA      <--
00029          6827 30 8D 0019    LEAX    ANALYS,PCR <--
00030          682B BF 01FC      STX     $1FC      <--
00031          682E 30 8D 0654    LEAX    BUFF,PCR
00032          6832 AF 8D 0426    STX     BUFF0,PCR
00033          6836 6F 8D 0421    CLR     CCNT,PCR
00034          683A 30 8D 005E    LEAX    MSG1-1,PCR
00035          683E BD 9FDF      JSR     $9FDF    <-- MSG PRINT
00036          6841 7E 948D      JMP     $948D    <-- BASIC HOT
00037          *
00038          6844 81 E8        ANALYS CMFA    #$E8    PSG ?
00039          6846 1027 0201 6A4B LBEQ    PSG
00040          684A 81 E9        CMFA    #$E9    MUSIC ?
00041          684C 27 6C 68BA    BEQ     MUSIC
00042          684E 81 EA        CMFA    #$EA    TEMPO ?
00043          6850 1027 016B 69BC LBEQ    TEMPO
00044          6854 81 EB        CMFA    #$EB    CRT ?
00045          6856 1027 026F 6AC9 LBEQ    CRT
00046          685A 81 EC        CMFA    #$EC    SCROLL ?
00047          685C 1027 0222 6A82 LBEQ    SCROLL
00048          6860 81 ED        CMFA    #$ED    KSYMBOL ?
00049          6862 1027 04C9 6D2F LBEQ    KSYMBOL
00050          6866 81 EE        CMFA    #$EE    KPRINT ?
00051          6868 1027 03F2 6C5E LBEQ    KPRINT
00052          686C 81 EF        CMFA    #$EF    COPY ?
00053          686E 1027 02B6 6B28 LBEQ    COPY
00054          6872 7E 9740      JMP     $9740    --> Syntax Error
00055          *
00056          6875 50          CODE FCC    'PSG?'    PSG for PSG CARD
00057          6878 4D          FCC    'MUSIC?'    MUSIC Music Play
00058          687D 54          FCC    'TEMPO?'    TEMPO for MUSIC

00059          6882 43          FCC    'CRT?'    CRT Display or Not
00060          6885 53          FCC    'SCROLL?'    SCROLL UP or DOWN
00061          688B 4B          FCC    'KSYMBOL?'    KSYMBOL
00062          6892 4B          FCC    'KPRINT?'    KPRINT KANJI Print to PRINTER
00063          6898 43          FCC    'COPY?'    COPY Window Copy
00064          689C 00          FCB     0
00065          *
00066          689D 0D          MSG1 FCB    13,10
00067          689F 45          FCC    'Expanded F-BASIC !!'
00068          68B2 00          FCB     0

```

## リスト2 FM-8F-BASIC EXPANDERダンプ・リスト

```

dd +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6800 17 08 1A 30 BC FB 9C 45 24 11 DC 45 93 3F 34 06 :30
6810 9F 45 1F 10 A3 E1 DD 3F BD 9A 14 86 08 B7 01 F9 :57
6820 30 8D 00 51 BF 01 FA 30 8D 00 19 BF 01 FC 30 8D :17
6830 06 8E AF 8D 04 26 6F 8D 04 21 30 8D 00 5E 8D 9F :72
6840 DF 7E 94 8D 81 E8 10 27 02 01 E9 27 6C 81 EA :89
6850 10 27 01 68 81 E8 10 27 02 02 6F 81 EC 10 27 02 :7C
6860 B1 ED 10 27 04 C9 B1 EE 10 27 03 F2 81 EF 10 27 :B4
6870 02 86 7E 97 40 50 53 C7 4D 55 53 49 C3 54 45 4D :5E
6880 50 CF 43 52 D4 53 43 52 4F 4C CC 48 53 59 4D 42 :5D
6890 4F CC 48 50 52 49 4E D4 43 4F 50 D9 00 0D 0A 45 :8A
68A0 78 70 61 6E 64 65 64 20 46 2D 42 41 53 49 43 20 :F9
68B0 21 21 00 00 00 00 00 00 00 00 9D D2 8D 9D 86 5D :EE
68C0 26 01 39 34 08 17 05 BF 12 D7 B9 A6 80 81 30 25 :15
68D0 1E 81 40 25 20 81 41 25 16 81 48 27 4D 25 69 81 :6D
68E0 4C 27 40 81 4D 27 3F 81 50 15 81 52 27 4E 0A :76
68F0 B9 26 D8 35 88 84 0F 31 8D 00 E6 31 A6 A6 A4 97 :63

```

Sum: DF 8B 8B F0 BF 30 5F 20 80 F9 B8 DD 3F E5 A5 F6 :50

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6900 B5 34 02 96 B4 27 12 81 01 27 10 81 02 26 0E 35 :13
6910 02 44 34 02 44 AB E0 20 06 64 E4 64 E4 35 02 97 :CF
6920 B6 20 CC 4F 20 06 86 20 02 86 40 97 B3 20 BF :CE
6930 A6 80 04 A4 03 97 B4 0A 27 B8 20 B2 E6 8D FF :28
6940 74 8D 6E 5A 26 FB 20 A7 D6 B6 D7 B8 D6 B5 D7 B7 :E5
6950 4A 84 07 48 48 9B B3 E6 84 C1 23 26 06 8B 02 30 :EA
6960 01 0A B9 31 8D 00 84 31 A6 A6 A0 5F 8D 28 A6 A4 :81
6970 5C 8D 23 86 0F C6 08 8D 1D 36 0D 88 27 06 0A :08
6980 B8 26 02 8D 07 0A B7 26 F0 16 FF 83 86 10 C6 08 :27
6990 8D 04 86 09 C6 0D 34 02 86 03 B7 FD 0D F7 FD 0E :75
69A0 7F FD 0D 35 02 B7 FD 0E 86 02 B7 FD 0D 7F FD 0D :54
69B0 39 34 10 8E 00 00 30 1F 26 FC 35 90 8D 00 D2 8D :8D
69C0 9E 49 BD A3 66 5D 26 03 7E 9B 03 86 08 3D ED BC :93
69D0 E3 86 FE C6 07 8D BF 4F C6 08 8D BA 86 14 5C 20 :FD
69E0 B5 06 0C 12 18 24 30 48 60 90 C0 AE 00 A4 00 9B :2A
69F0 00 92 00 25 01 15 01 05 01 F6 00 E8 00 DB 00 DB :68

```

Sum: 61 82 09 BD 7A 8C B9 0A C4 A1 F4 52 3B D9 1D 21 :9F

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
AA00 00 CD 00 C3 00 88 00 00 00 00 00 57 00 52 00 4D :3E
AA10 00 49 00 92 00 8A 00 00 82 00 7B 00 74 00 6D 00 :80
AA20 00 66 00 61 00 5C 00 00 00 00 00 2B 00 29 00 26 :9D
AA30 00 24 00 49 00 45 00 41 00 3D 00 3A 00 36 00 36 :D6
AA40 00 33 00 30 00 2E 00 00 00 00 00 9D D2 8D 9E 8E :E1
AA50 8D A3 66 9E 4B 8C 00 10 25 03 7E 9B 03 34 04 1F :E6
AA60 10 35 02 16 FF 30 00 00 3F 59 41 4D 41 55 43 48 :D3
AA70 49 93 D3 8F 90 FC 0D 1D 00 00 00 FD D0 1D FD D4 :72
AA80 0E 39 9D D2 8D 9E 86 BD A3 66 C1 01 27 0A 5D 27 :D4
AA90 03 7E 9B 03 86 C3 20 02 86 83 A7 8C DB DC 4B ED :B5
AAA0 8C 07 30 8C C1 BF 05 A2 8E 05 A0 86 10 A7 84 CC :06
AAB0 00 8D 0E 04 0E DE 00 00 3F 59 41 4D 41 55 43 48 :A4
AAC0 49 93 D3 8F 90 0D 04 08 39 9D D2 81 97 27 07 81 :19
AAD0 D5 27 07 7E 97 40 86 B6 20 02 86 87 A7 8C E6 9D :A9
AAE0 D2 30 8C D2 20 BF 11 00 04 4B 00 00 80 00 80 00 :9C
AAF0 1B 00 00 00 00 00 00 00 07 01 02 03 04 05 06 :37

```

Sum: BE 36 F6 B6 33 C6 E6 0F B7 49 61 CC 7A 9A 43 23 :35

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
AB00 07 9D D2 BD 97 2F BD 9E 86 34 04 BD 97 2C 35 04 :CB
AB10 C1 19 25 03 7E 9B 03 9E 4B 8C 00 50 24 F6 39 00 :36
AB20 00 00 00 00 00 00 00 00 8D D7 E7 8C F7 1F 10 E7 :E4
AB30 8C F1 9D D8 81 DA 27 03 7E 97 40 8D C4 E7 8C E6 :76
AB40 1F 10 E7 8C 0E 6F 8C D9 6F 8C D7 9D 08 27 14 81 :59
AB50 2C 26 E5 9D D2 8D 9E 8F 8C 02 25 03 7E 9B 03 :E2
AB60 AF 8C BE A6 8C BD A1 8C BC 24 0D 34 02 A6 8C B5 :1F
AB70 A7 8C BD 35 02 A7 8C AD A6 8C A9 A1 8C AB 24 0D :DB
AB80 34 02 A6 8C A1 A7 8C 9C 35 02 A7 8C 99 A6 8C 93 :A0
AB90 A0 8C 92 4C C6 08 3D E3 8C 87 ED 8C 82 A6 8C 85 :BD
ABA0 A7 8C 84 86 1B 8D 6D 86 4C 8D 5C A6 8D FF 71 8D :30
ABB0 56 A6 8D FF 6A 8D 50 10 AE BD FF 65 27 07 4F 8D :88
ABC0 46 31 3F 26 F9 8D 43 A6 8D FF 58 A1 8D FF 56 27 :D9
ABD0 06 C6 8D FF 50 20 EE A6 8D FF 4C A7 8D FF 46 A6 :F9
ABE0 8D FF A1 A1 8D FF 3F 27 14 6C 8D FF 39 86 1B 8D :D3
ABF0 1B 86 A1 BD 12 86 08 BD 0E BD 0A 20 A6 86 1B 8D :30

```

Sum: B5 D7 65 4C AA 2F 2F F5 30 06 62 47 A7 77 13 30 :7A

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
AC00 06 86 32 8D 02 86 0A 7E D8 AB 8E 04 40 34 10 31 :22
AC10 8D FE D3 C6 1B A6 A0 A7 B0 5A 26 F9 35 10 A6 8D :9D
AC20 FF 03 C6 08 3D ED 0B C3 00 07 ED 0F A6 8D FE F6 :F2
AC30 C6 08 3D ED 0C 03 00 07 ED 88 11 9D DE 8E 04 4C :AE
AC40 C6 08 34 14 C6 08 68 80 49 5A 26 FA BD DB AB 35 :01
AC50 14 5A 26 EE 39 16 00 04 40 00 00 00 06 86 9D D2 :78
AC60 B1 DC 27 26 BD 9E 8F AF 9C F2 AE 8C EF 30 02 AF :DB
AC70 8C EA 6C 8C E6 A6 8C E3 81 30 26 02 8D 0E 9D DB :52
AC80 26 01 39 81 2C 27 D7 7E 97 40 9D D2 6D 8C CC 26 :8A
AC90 05 86 0A 7E D8 AB 8D 7E CE 04 40 8D 35 86 1B 8D :AA
ACA0 2E 86 4A 8D 2A 86 01 8D 26 86 0D 8D 22 8D 67 CE :F3
ACB0 04 42 8D 1E 86 1B 8D 17 86 4A 8D 13 86 23 8D 0F :EB
ACC0 86 0D 8D 0B 30 8D 01 D8 AF 8C 91 6F 8C BD 39 7E :CC
ACD0 DB AB 31 BD 01 CA E6 8C B2 34 44 EC A1 ED 8D FF :7B
ACE0 7B 30 8D FF 70 9D DE 34 40 8D 14 35 40 33 41 8D :AA
ACF0 0E C6 04 4F 8D D9 5A 26 FA 35 44 5A 26 DB 39 C6 :DA

```

Sum: 80 B1 5E 8C EB 7B 49 63 67 A3 50 1A 7D 45 B7 EE :0B

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
AD00 08 34 44 C6 08 68 C4 49 33 44 5A 26 FB BD DB AB :EF
AD10 35 44 5A 26 EC 39 86 1B 8D 85 86 4C BD B1 86 14 :AB
AD20 E6 8D FF 37 3D 34 02 1F 9B 8D A4 35 02 20 A0 9D :98
AD30 D2 BD E2 4D 9E C5 D6 CB BF 05 04 F7 05 06 BD 9E :E4
AD40 A8 F7 05 0A BD 9E 8A F7 05 86 05 0A 3D 4D 26 F1 :F1
AD50 19 86 20 3D 27 14 BD 92 86 86 05 0B 48 4B 48 :F2
AD60 B8 05 06 25 05 4A B1 C8 25 05 C6 05 7E 92 AB 87 :EA
AD70 05 07 B6 01 E5 97 FB 0F C2 9D 8B 81 3B 27 28 BD :48
AD80 97 32 81 2C 27 0B E0 D1 E6 1A D7 9F 9D 8B 81 3B :27 :8F
AD90 16 BD 97 32 BD E0 D1 24 03 7E 97 40 D7 C2 9D D2 :8E
ADA0 B1 3B 27 03 7E 97 40 9D D2 26 01 39 B6 05 0A C6 :95
ADB0 10 3D FE 05 04 DF C5 33 CB 11 83 02 80 22 35 FF :62
ADC0 05 04 33 5F DF C9 FC 05 06 0F C7 0F 8C 97 C8 D7 :30
ADD0 CC BD 97 64 BD 9E 92 DE 3D BD 25 86 9F 97 F8 7F :D1
ADE0 01 A5 DF F5 DE 3D DF F3 BD EB 27 9D DB 27 0D BD :9C
ADF0 97 32 20 B8 9D DB 27 04 9D D2 26 FC 39 FD 05 0A :17

```

Sum: FF 4A 66 B3 1A CA 0C 5F E0 D3 30 7A 57 8E 0C B4 :F3

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
AE00 34 08 86 05 1F 88 1F 10 8E 05 A0 84 7F C4 7F ED :06
AE10 04 86 16 A7 84 DC 0A 3D 10 83 00 01 23 5F CC 05 :D5
AE20 10 ED 02 BD F2 DB 1F 01 10 8E 00 08 96 08 97 09 :8D
AE30 86 10 97 0C A6 84 97 0B 96 0A 08 08 1F A9 20 02 :9C
AE40 1F 9A 69 C4 31 3F 26 06 33 41 10 8E 00 08 4A 26 :0C
AE50 EF 0A 0C D6 0C 27 08 C1 08 26 D0 30 01 20 05 0A :12
AE60 09 27 04 30 1F 20 C9 30 01 8C 05 25 8E 1F 20 :80
AE70 C1 08 27 05 68 CA 5A 26 FB 33 41 35 88 9F 02 BD :7B
AE80 F2 D8 33 C8 14 35 88 34 04 1F 50 80 06 1F BB 35 :A2
AE90 84 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :84

```

Sum: 1C 36 08 0C 13 42 B8 A7 7F 65 2B 38 0B CB CD 3F :43

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
AF00 30 8C FB 9C 45 24 16 DC 45 93 3F 34 06 9F 45 1F :02
AF10 10 A3 E1 DD 3F BD 94 14 BD 03 7E 94 8D 86 7E B7 :FF
AF20 02 69 30 8D 00 15 BF 02 6A 30 BD 0F 07 C6 10 6F :80
AF30 80 5A 26 FB 30 8D 0C CD 7E 9F DF 32 62 34 FF 1F :73
AF40 50 8B 0F 1F 8B 10 FF 03 16 10 FE 03 16 0F 31 17 :3A
AF50 05 9E 86 2A 17 05 D4 17 05 34 86 04 3D 27 EA 81 :1C
AF60 2A 10 26 04 AB 17 04 32 17 04 20 17 04 2C 81 42 :A1
AF70 10 27 03 8B 81 44 10 27 01 25 81 47 27 29 B1 49 :C9
AF80 27 34 81 4B 27 48 81 4C 10 27 00 A1 81 4D 10 27 :40
AF90 02 3B 81 52 10 27 03 08 81 5B 27 5F 10 21 05 AD :91
AFA0 86 07 17 05 86 20 A2 17 03 E1 27 05 17 03 F6 ED :15
AFB0 6A 17 04 A4 35 FF 17 03 EC 1F 01 17 03 BF 17 04 :77
AFC0 97 A6 9F 00 D9 27 DE A7 80 17 03 CE 20 F3 8D 14 :7D
AFD0 DC 48 4D 26 CB 17 0D 80 9E 44 E7 80 9C 46 25 FA :47
AFE0 27 FB 20 C1 17 03 BE DD 44 17 03 B7 DD 46 10 93 :90
AFF0 44 27 02 25 AB 17 03 AB DD 4B 39 8D E7 17 04 58 :47

```

Sum: 4B E9 1B 2B DA D9 3C 4F AC 0B F3 1C A5 70 D7 45 :AC

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
B000 9E 44 9C 48 25 0F 10 9E 48 A6 80 A7 A0 9C 46 25 :64
B010 FB 27 A6 20 CD CC 46 93 44 D3 48 1F 02 9E 46 A6 :C1
B020 84 A7 A4 30 1F 31 3F 9C 44 24 F4 20 B5 BD 2E 0D :23
B030 32 10 26 FF 6B DE 44 0D 33 27 09 C6 0F 8D 10 5A :30
B040 26 FB 20 9E 8D 09 11 93 46 25 F9 27 F0 20 F3 17 :C5

```

## リスト 2 F-BASIC EXPANDER ダンプ・リスト

```

7150 04 FB FF 03 14 7D 03 13 10 26 FF 44 39 0F 32 0F :AA
7160 33 17 03 27 26 08 BE 03 14 9F 44 03 33 39 17 03 :E3
7170 34 DD 44 17 03 15 27 F3 17 03 28 DD 46 17 03 08 :28
7180 27 EB 17 02 FB 17 03 03 27 E3 81 41 26 07 97 32 :02
7190 17 03 07 20 F0 81 50 10 26 FF 05 97 31 20 F1 8D :A2
71A0 BC 0F 4C DE 44 0D 33 27 09 C6 08 8D 17 5A 26 FB :96
71B0 20 9B 8D 10 11 93 46 25 F9 27 F7 0D 4C 27 BE 17 :A3
71C0 00 D1 20 B9 34 04 0D 31 26 2D 86 01 97 32 B6 00 :49
71D0 C3 81 28 26 22 1F 30 17 04 3E C6 08 8D 76 33 5B :8B
71E0 17 04 5D C6 08 17 00 90 5A 26 FA 17 04 02 7D 03 :04
71F0 13 10 26 FE AB 35 84 0D 4C 26 15 30 8D 0B 33 17 :51

```

Sum: E4 0A B4 F9 8C 44 5F BA A3 37 09 B9 7E 30 DE A9 :25

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
7200 04 42 17 03 EB 30 8D 0D 44 C6 11 6F 80 5A 26 FB :9A
7210 1F 30 17 04 03 C6 10 8D 3B 86 3A 17 04 0D 96 4D :D6
7220 17 03 F9 0D 32 27 0F 17 04 16 17 04 13 33 50 C6 :30
7230 10 8D 45 5A 26 FB 17 03 B7 0C 4C 96 4C 81 10 26 :1F
7240 AD 8D 50 0F 4C 0D 31 27 06 17 03 A4 17 03 A1 17 :E0
7250 03 9E 20 9A 0F 4D 31 8D 0C F4 17 03 E3 A6 C4 9B :77
7260 4D 97 4D A6 C4 AB A4 A7 A0 A6 C0 17 03 AE 17 03 :79
7270 CF 5A 26 E9 FF 03 14 39 A6 C0 81 20 25 10 81 7F :C3
7280 27 0C 81 FB 25 0A 81 FF 27 04 0D 31 27 02 86 2E :A1
7290 16 03 98 0F 4D C6 39 86 2D 17 03 8F 5A 26 FB 30 :10
72A0 8D 0A C8 17 03 9E 30 8D 0C A4 C6 10 A6 84 9B 4D :6C
72B0 97 4D A6 80 17 03 65 17 03 86 5A 26 EF B6 3A 17 :6F
72C0 03 69 96 4D 17 03 55 16 03 26 17 01 BE 27 06 17 :17
72D0 01 D3 FD 03 14 BE 03 14 1F 10 17 03 3B 17 03 60 :8B
72E0 A6 84 17 03 37 86 2D 17 03 41 8D 31 48 48 48 48 :67
72F0 97 4C 8D 29 9A 4C A7 80 0F 03 14 1F 10 C4 0F 26 :A4

```

Sum: B8 90 0D C0 EC 24 5B 32 D9 9E 08 4B 6C FE CC 0F :BB

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
7300 17 86 07 17 03 25 86 3A 17 03 20 30 10 C6 10 4F :42
7310 AB 80 5A 26 FB 17 03 04 17 02 D5 20 B8 17 02 4E :F1
7320 1F 89 17 01 82 25 07 34 02 17 02 FA 35 82 86 0A :2E
7330 C1 2E 27 F3 C1 0D 27 23 C1 2D 27 1E C1 2B 27 19 :80
7340 C1 2C 27 14 C1 1C 27 0F C1 1D 27 12 C1 1F 27 12 :68
7350 C1 1E 27 14 16 FD 49 4C 4C 4C 4C 4C 4C 4C 4C 4C :2A
7360 20 02 30 01 32 62 20 90 9F 44 30 1F 0F 4C B6 00 :DA
7370 C3 81 28 27 06 1F 10 C4 F0 20 04 1F 10 C4 F8 1F :AA
7380 03 17 02 6C 17 FE 3D 17 02 04 17 00 FE 27 07 17 :51
7390 01 83 9E 4A 20 04 9E 4A 8F 03 14 32 62 16 FF 35 :F0
73A0 1F 43 31 8D 07 57 C6 04 34 8D 3B A6 C4 17 02 :C8
73B0 6B 8D 3E 27 0A 17 00 ED 4D 10 26 FC E3 E7 C4 33 :AB
73C0 41 35 04 5A 26 E2 C6 04 34 04 8D 18 EC C4 17 02 :4C
73D0 47 8D 1E 27 05 17 00 CD ED C4 33 42 35 04 5A 26 :E1
73E0 E7 16 FC C1 A6 A0 17 02 42 A6 A0 17 02 3D 16 02 :0F
73F0 4F 86 2D 17 02 35 17 02 05 17 01 92 16 00 8C 86 :40

```

Sum: 53 52 9F 3E 9B 16 EC 65 37 B6 04 6D E0 6F B8 41 :2A

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
7400 7E B7 01 D7 30 8D FC 37 BF 01 D8 30 8D 08 25 86 :08
7410 01 34 02 17 02 10 17 02 27 6D 84 26 30 C6 04 86 :37
7420 2A 17 02 07 5A 26 FB 8D 47 27 16 8D 7B ED 01 1F :E5
7430 02 A6 A4 A7 03 86 3F A7 A4 A1 A4 10 26 FC 61 A7 :B5
7440 84 30 04 35 02 4C 91 05 26 C7 16 FC 58 EC 01 17 :1C
7450 01 C6 8D 1C 27 EB 81 2A 26 09 6F 84 A6 03 A7 9B :37
7460 01 20 DE 8D 40 34 06 A6 03 A7 9B 01 35 06 20 8D :07
7470 86 2D 17 01 86 17 01 86 17 01 13 20 0E 8D 0C 10 :21
7480 27 FC 1D 81 2C 10 26 FC 17 20 0F A6 9F 00 D9 27 :AA
7490 0B 81 20 26 04 8D 03 20 F2 39 7C 0D A6 26 03 7C :A9
74A0 00 D9 39 8D 0B 0F 4A 0F 4B 8D E0 10 27 FB F1 8D :47
74B0 26 10 25 FB EB 86 04 97 4C A6 9F 00 D9 8D 18 25 :96
74C0 13 C6 04 08 4B 0F 4A 5A 26 F9 9A 4B 97 4B 8D C4 :1A
74D0 0A 4C 26 E5 CD 4A 39 81 30 25 12 81 3A 25 0A 81 :13
74E0 41 25 0A 81 47 24 05 80 07 80 30 39 43 39 A6 80 :73
74F0 8D E5 25 10 4B 48 48 48 34 02 A6 80 8D D9 25 02 :80

```

Sum: F7 6D 23 28 57 BC 9A 2D 68 DA D2 CF B6 6C A6 70 :A4

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
7500 AA 04 32 61 39 8D E7 25 06 1F 89 8D E1 1E 89 39 :EF
7510 8D 03 16 FB 90 8E 04 3D 8B E8 10 25 FB 82 1F 02 :48
7520 A6 80 81 2D 27 1B 81 20 10 26 FB 74 8D C0 25 09 :D7

```

```

7530 A7 A0 30 01 8C 04 71 25 F3 8D 8B 86 04 3D 26 D0 :76
7540 39 A6 84 81 20 26 04 30 01 20 F6 8C 04 4D 24 E9 :5F
7550 8D 9C 25 E5 A7 A0 20 E9 34 12 30 8D 07 BE 17 00 :62
7560 E3 8D 0B 17 00 C5 81 59 10 26 FB 34 35 92 8D 14 :FE
7570 34 7C 4F 1F 8B 8D D3 36 35 7C 34 7E 4F 1F 8B 8D :88
7580 DC F3 35 FE 34 7E 4F 1F 8B 8D DC FA 35 FE 34 76 :1D
7590 8D F2 7F 03 13 CE DC 0C 8D DB FF BD F2 8D 8D 18 :8D
75A0 8D 8D 8D 4C 35 F6 34 76 7F 03 13 CE DC 14 BD DB :FE
75B0 FF BD F2 8D 8D 02 35 F6 7D 03 13 27 0B 4F 1F 8B :FB
75C0 1A 01 7E 94 8D 10 BE 05 A4 8E 04 3D BF 00 D9 10 :A8
75D0 8C 00 04 27 10 10 8C 00 01 27 0D 31 3C A6 04 A7 :56
75E0 80 31 3F 26 FB 6F 84 39 A6 84 81 01 27 CF 20 F5 :F1
75F0 86 04 8D 37 86 0D 8D 33 6D 8D 09 35 26 E9 BD C8 :87

```

Sum: 02 08 7D 63 F2 62 44 57 0C F5 F0 F7 4F F0 9D 4A :E7

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
7600 41 34 16 86 11 F6 01 E5 FD 01 04 86 02 B7 01 03 :43
7610 8E 05 A0 8D F2 D8 35 96 8D 02 1F 9B 34 02 47 47 :8F
7620 47 47 8D 02 35 02 84 0F 17 03 A0 34 1E 5F 1F 9B :0C
7630 6D 8D 0B FD 26 05 BD 03 52 35 9E BD DB A6 35 9E :EF
7640 86 20 8D 07 A6 80 27 04 8D E1 20 FB 39 34 37 17 :3F
7650 00 81 30 8D 0B AA EC 81 8B BE 86 2D BD CD 0D 29 :EB
7660 27 06 EC 81 8D B2 20 8B A6 81 8D 8D D2 8D D0 :21
7670 8D CE C6 07 96 28 27 0D 34 02 A6 80 8D 9E 35 02 :D8
7680 5A 5A 4A 26 F1 5D 27 05 8D B6 5A 26 FB A6 80 81 :0B
7690 20 27 03 8D 95 31 BD 0B 73 34 20 AC E1 26 ED :F5
76A0 C1 04 27 13 A6 80 81 20 27 04 17 FF 7E 5C C1 04 :A6
76B0 27 05 8D 8C 5C 20 F7 8D 87 8D 85 A6 80 81 20 27 :CC
76C0 03 17 FF 67 31 8D 0B 5F 34 20 AC E1 26 ED 17 FF :AF
76D0 1F 35 37 30 8D 0B 29 86 20 C6 27 A7 80 5A 26 FB :2E
76E0 86 01 97 28 8D 0E 96 27 4A 48 30 8D 04 47 EC 86 :AA
76F0 1F 51 6E 8B 31 8D 0B 0E EF A1 0F 29 E6 C0 E7 A0 :2C

```

Sum: E6 AA 09 A9 2B 9B 70 4A B7 E6 76 8D 41 E3 39 4E :0D

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
7700 01 10 26 0A 0D 29 10 26 01 0C 0C 29 20 EE C1 11 :8F
7710 26 0C 0D 29 10 26 00 FE 0C 29 20 EE 0C 29 30 8D :B1
7720 04 E4 3A E6 84 C1 88 10 27 00 A6 0D 29 10 27 00 :4F
7730 EE C1 54 26 1A D6 03 54 10 24 00 95 30 8D 04 C6 :0C
7740 C6 81 3A 0A 29 26 03 86 44 8C 86 55 5F 20 68 C1 :B6
7750 57 26 0D 0A 29 26 03 86 32 8C 86 33 C6 03 20 57 :23
7760 96 03 84 F0 81 20 26 19 0A 29 10 26 00 A6 96 03 :95
7770 84 0F 10 27 00 9E CC 4C 42 DD 07 C6 0A 86 20 20 :3C
7780 36 96 03 81 80 25 44 96 03 84 0F 81 0E 24 16 81 :B5
7790 0C 26 3E E6 84 C1 18 26 38 0A 29 26 03 86 59 8C :D8
77A0 86 53 5F 20 12 96 03 84 F0 81 C0 24 03 86 59 8C :4A
77B0 86 53 5F 0A 29 26 5D 0C 29 97 0D 07 27 4F E6 84 :7E
77C0 30 8D 03 87 3A 31 8D 07 40 A6 80 2B 06 A7 A0 20 :44
77D0 F8 20 43 84 7F A7 A0 0D 27 26 0D 29 26 04 96 1E :A8
77E0 02 20 02 96 03 84 F0 26 04 86 06 20 15 44 44 44 :E8
77F0 44 84 03 26 10 86 04 D6 0D C1 41 27 05 C1 42 27 :C6

```

Sum: CC 2D E6 C2 99 74 A6 55 D2 30 EA 4E 38 54 32 DD :7E

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
7800 01 4C 97 27 39 81 01 27 E0 81 02 26 03 86 08 8C :93
7810 86 07 20 EE 0C 29 C6 0B D7 27 4F C6 8B 20 A1 D6 :03
7820 02 C1 80 25 48 C1 C0 25 03 86 42 8C 86 41 C4 0F :47
7830 C1 03 26 04 86 44 20 31 C1 0C 25 2D C1 0D 26 12 :2E
7840 E6 84 C1 64 27 EE C1 91 27 02 5F 8C C6 09 86 20 :7F
7850 20 18 D6 02 C1 C0 24 04 86 58 20 0D D6 02 C4 0F :5F
7860 C1 0E 24 03 86 44 8C 86 55 1F FF 4C C4 F0 C1 :6C
7870 60 24 0F C1 40 25 0E 26 03 86 41 8C 86 42 C6 03 :D4
7880 20 E8 5F 20 C9 C1 10 24 02 20 C3 C1 20 26 06 86 :8D
7890 42 97 8F 20 B7 C1 30 26 21 D6 02 C4 0F C1 04 24 :84
78A0 04 C6 08 20 A9 C1 08 24 04 C6 01 20 A1 C1 0C 26 :07
78B0 03 C6 04 8C C6 03 86 20 20 B0 D6 02 C4 0F C1 0E :12
78C0 25 04 C6 02 20 8B C1 0D 27 EA C1 0A 25 06 C6 04 :38
78D0 86 43 20 96 C1 09 27 DC C1 06 25 D8 C6 0A 20 D6 :D6
78E0 E6 C0 D7 04 0F 2C 34 04 54 54 54 54 31 8D 06 1E :26
78F0 31 21 8D 19 C6 2C E7 A0 35 04 30 01 C4 0F 8D 0D :48

```

Sum: 9C 18 E4 09 66 F5 F7 E4 38 2D 9A A7 E4 68 E3 59 :FF

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
7900 04 2C 25 01 39 CC 2A 2A DD 0F DD 12 39 34 10 30 :37

```

## リスト 2 F-BASIC EXPANDER ダンプ・リスト

```

7910 8D 01 FA C1 06 25 02 0C 2C C1 0C 25 02 C6 06 58 :C6
7920 3A EC 81 ED A1 35 90 A6 C0 97 04 31 8D 01 CE 30 :88
7930 8D 05 DB C6 08 D7 20 44 34 02 2A 21 EC A1 34 02 :C1
7940 96 2D 81 02 26 0E 96 02 84 0F 81 06 25 06 35 02 :8E
7950 86 53 20 02 35 02 ED 81 86 2C 87 80 8C 31 22 35 :8D
7960 02 0A 2D 26 D2 86 20 A7 82 39 A6 C0 97 04 8D 46 :0D
7970 8D 13 20 18 8D FA D0 11 A6 C0 97 05 0C 28 20 EE :D8
7980 86 23 97 0F 20 E4 86 23 97 0F 20 E8 34 02 86 2A :8A
7990 97 10 35 82 E6 C0 D7 04 1D DF 2A D3 2A 34 04 8D :C7
79A0 15 DD 11 35 02 8D 0F DD 13 20 E1 EC C1 DD 04 0C :61
79B0 28 2D E6 0A 28 39 8D 08 1E 89 8D 0F 1E 89 20 08 :43
79C0 1F 89 84 0F 1E 89 44 44 44 44 39 81 0A 25 02 88 :68
79D0 07 8B 30 39 86 2D 97 0F 39 A6 C4 84 60 27 11 81 :94
79E0 20 27 0A 81 40 27 03 86 53 8C 86 55 8C 86 59 8C :73
79F0 86 58 97 18 86 2C 97 15 A6 C0 97 04 2B 14 84 1F :CE

```

Sum: 79 7E 81 68 3C FA D7 55 8A 6A 48 E8 66 B1 BA A4 :AB

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
7A00 1F 89 58 58 58 2A 08 8D C8 50 54 54 1F 98 16 :53
7A10 FF 5C 84 0F 31 8D 01 0D E6 A6 4F 1F 52 6E AB 86 :A5
7A20 2B 97 19 17 00 96 10 26 00 8D 39 CC 2B 28 DD 19 :9C
7A30 20 0B 86 2D 97 16 20 EB CC 2D 2D DD 16 20 7D 86 :D2
7A40 42 97 12 20 77 86 41 20 F8 A6 C0 97 05 2A 05 C6 :58
7A50 2D 07 0F 40 17 FF 17 0C 2B 20 61 D7 05 EC C1 DD :98
7A60 05 2A 0D 50 26 01 4A 43 34 02 86 2D 97 0F 35 02 :06
7A70 34 04 17 FF 41 DD 11 35 02 17 FF 3A DD 13 17 FF :0A
7A80 0B 0C 28 2D D2 86 40 20 88 8D 17 8D CA E6 C0 D7 :48
7A90 05 1D 16 FF 04 8D 0B 8D BE 0C 28 EC C1 DD 05 17 :F8
7AA0 FE F7 CC 50 43 DD 18 39 EC C1 DD 05 8D C2 27 07 :8E
7AB0 86 2D 97 18 97 15 39 CC 2A 2A 2D E9 34 02 96 04 :33
7AC0 85 10 26 02 35 82 86 58 97 0E 86 5D 97 18 35 82 :46
7AD0 86 27 97 0E 97 10 D6 02 8D 0F D7 0F 0A 28 DD 29 :85
7AE0 28 0C 39 C1 20 2B 0A C1 60 2B 02 C6 2E 39 97 12 :9F
7AF0 97 14 86 2C 97 11 D6 03 8D E9 D7 13 39 43 43 41 :3E

```

Sum: 6D BA DD DE 4B 99 C2 22 70 3E 21 9D B9 56 4D D6 :45

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
7B00 20 42 20 44 50 58 20 59 20 55 20 50 43 44 20 58 :CB
7B10 20 59 20 55 20 53 20 50 43 2A 2A 2A 2A 21 20 42 :5F
7B20 20 43 43 44 50 02 0E 15 18 9F 22 28 9A 2C 40 9A :03
7B30 6B 6C 78 9A 8B 02 35 01 EE 02 C1 02 8E 02 94 02 :82
7B40 78 02 82 02 87 02 A2 02 89 03 DE 41 42 D8 41 44 :05
7B50 C3 41 44 C4 41 4E C4 41 53 CC 41 53 D2 42 49 D4 :84
7B60 43 4C D2 43 4D D0 43 4F CD 44 41 C1 44 45 C3 45 :F7
7B70 4F D2 45 58 C7 49 4E C3 4A 4D D0 4A 53 D2 4C 53 :54
7B80 CC 4F 53 D2 4D 55 CC 4E 45 C7 4E 4F D0 52 4F CC :DF
7B90 52 4C D2 52 54 59 52 54 D3 53 42 C3 53 45 D8 53 :76

```

```

7BA0 55 C2 53 57 C9 54 46 D2 54 53 D4 4C C4 4F D2 53 :F5
7BB0 D4 43 C3 43 D3 45 D1 47 C5 47 D4 48 C9 4C C5 4C :9B
7BC0 D3 4C D4 4D C9 4E C5 50 CC 52 C1 52 CE 53 D2 56 :E6
7BD0 C3 56 D3 43 57 41 C9 53 59 4E C3 4C 42 53 D2 4C :4C
7BE0 42 52 C1 4C 45 41 D8 4C 45 41 D9 4C 45 41 D3 4C :9B
7BF0 45 41 D5 50 53 48 D3 50 53 48 B5 50 55 4C D3 50 :ED

```

Sum: F9 80 50 C2 7C E7 E8 0E 7D 5D C7 23 9A 49 B5 E2 :22

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
7C00 55 4C D5 46 43 C3 3C 8B 8B 18 36 8B 45 0F 0C 42 :19
7C10 21 8B 2A 5D 2D 15 8B 8B 3F 8C 8B 8B 94 90 8B 1E :47
7C20 62 8B 09 51 27 5A 7E 80 7A 66 6B 7A 6A 84 86 :93
7C30 7C 78 6C 76 6E 72 98 9C A0 A4 AB 80 AC BA 88 4E :E9
7C40 00 48 8B 39 8B 57 3C 8B 8B 18 36 8B 45 0F 0C 42 :6F
7C50 21 8B 2A 5D 8B 15 3C 8B 8B 18 36 8B 45 0F 0C 42 :84
7C60 21 8B 2A 5D 8B 15 3C 8B 8B 18 36 8B 45 0F 0C 42 :84
7C70 21 8B 2A 5D 2D 15 3C 8B 8B 18 36 8B 45 0F 0C 42 :F9
7C80 21 8B 2A 5D 2D 15 54 18 4E 54 09 12 60 8B 24 03 :0A
7C90 62 06 18 91 60 8B 54 18 4E 54 09 12 60 64 24 03 :3D
7CA0 62 06 18 30 60 64 54 18 4E 54 09 12 60 64 24 03 :8B
7CB0 62 06 18 30 60 64 54 18 4E 54 09 12 60 64 24 03 :88
7CC0 62 06 18 30 60 64 54 18 4E 54 09 12 60 8B 24 03 :8E
7CD0 62 06 60 8B 60 8B 54 18 4E 06 09 12 60 64 24 03 :5E
7CE0 62 06 60 64 60 64 54 18 4E 06 09 12 60 64 24 03 :B6
7CF0 62 06 60 64 60 64 54 18 4E 06 09 12 60 64 24 03 :B6

```

Sum: 86 86 24 B8 27 B3 9A 2C 57 93 1C F8 13 C1 50 51 :FB

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
7D00 62 06 60 64 60 64 0D 0A 45 78 70 61 6E 64 65 64 :30
7D10 20 4D 4F 4E 49 54 4F 52 20 21 21 00 41 72 65 20 :E2
7D20 79 6F 75 20 73 75 72 65 20 3F 20 28 59 2F 4E 29 :E2
7D30 20 00 41 64 64 20 2B 30 20 2B 30 20 2B 32 20 :DD
7D40 2B 33 20 2B 34 20 2B 35 20 2B 36 20 2B 37 20 2B :AB
7D50 3B 20 2B 39 20 2B 41 20 2B 42 20 2B 43 20 2B 44 :F2
7D60 20 2B 45 20 2B 46 20 53 75 6D 00 0D 0A 53 75 6D :C2
7D70 3A 20 00 4C 61 73 74 20 56 65 72 73 69 6F 6E 20 :14
7D80 3B 32 F2 3F 33 32 35 F4 20 48 45 52 4F 2D 30 39 :C5
7D90 41 53 53 45 4D 42 4C 45 52 20 28 43 29 20 42 59 :0D
7DA0 20 45 54 46 20 4C 61 62 6F 72 61 74 6F 72 79 20 :5E
7DB0 20 20 44 65 73 69 67 6E 65 64 20 42 79 20 20 20 :9E
7DC0 20 20 54 2E 4B 61 73 61 73 61 68 75 20 20 20 20 :76
7DD0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
7DE0 20 56 65 72 73 69 6F 6E 20 31 2E 30 35 20 20 :4A
7DF0 42 41 53 49 43 20 45 58 50 41 4E 44 45 52 00 20 :F9

```

Sum: 13 01 DE 16 34 64 5E E4 F4 4B 79 B9 03 BA C3 FB :CB

## FM-7 EX-MOM ダンプ・リスト

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6000 30 8C FB 9C 45 24 16 DC 45 93 3F 34 06 9F 45 1F :02
6010 10 A3 E1 0D 3F 8D 8F 51 8D 03 7E 8F E1 86 7E 87 :86
6020 02 69 30 8D 00 15 BF 0E 6A 30 8D 0F 07 C6 10 5F :80
6030 8C 5A 26 FB 30 8D 0C 0E 7F 98 8B 32 62 34 FF 1F :8D
6040 50 8D 0F 1F 8B 8E 6A 30 1F AF 6A 10 FF 03 16 10 :4C
6050 FE 03 16 0F 31 17 05 B1 86 2A 17 05 EC 17 05 47 :3F
6060 B6 04 3D 27 EA 81 26 10 26 04 AF 17 04 36 17 04 :08
6070 24 17 04 36 81 42 16 07 03 86 81 44 10 27 01 27 :16
6080 81 47 27 2B 81 49 27 76 81 4B 27 4A 81 4C 10 27 :82
6090 06 A3 81 4D 10 27 02 43 81 52 10 27 03 0F 81 5B :E2
60A0 27 61 81 51 10 27 03 D7 86 07 17 05 9C 20 A0 17 :87
60B0 03 63 27 05 17 03 FB ED 6A 17 04 A6 35 FF 17 03 :8A
60C0 EE 1F 01 17 03 C1 17 04 99 A6 9F 00 D9 27 DE A7 :67
60D0 80 17 03 0D 20 F3 8D 14 DC 48 4D 26 CB 17 04 82 :10
60E0 9E 4A E7 80 9C 46 25 FA 27 FB 26 C1 17 03 0D DD :01
60F0 44 17 03 B9 DD 46 10 93 44 27 02 25 AB 17 03 AD :E1

```

Sum: E5 5A D6 74 2F E5 16 1B 5A 8C 36 9C 9A 6B F2 32 :19

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6100 DD 4B 39 8D E7 17 04 5A 9E 44 9C 4B 25 0F 10 9E :EF
6110 4B 46 80 A7 0A 9C 46 25 FB 27 F6 20 0D DC 46 93 :73
6120 4A D2 4B 1F 02 9E 46 A6 84 A7 44 30 1F 31 3F 9C :24
6130 44 24 F4 20 85 8D 2E 0D 32 10 26 FF 68 DE 44 0D :FA
6140 33 27 09 C6 0F 8D 10 5A 2E 3B 20 9E 8D 09 11 93 :48
6150 46 25 F9 27 F7 20 F2 17 05 15 FF 03 14 7D 03 13 :6F
6160 10 26 FF 44 79 3F 32 0F 35 17 03 29 26 8E 03 :67
6170 14 9F 44 03 39 17 03 36 DD 44 17 03 17 03 73 :22
6180 17 03 2A DD 46 17 03 0D 27 EB 17 02 FA 17 03 05 :D2
6190 27 F3 81 41 26 07 32 17 03 09 20 F0 81 50 10 :06
61A0 2A FF 05 97 31 20 F1 8D 0C 0F 4E 4A 0D 33 27 :30
61B0 9F C6 08 8D 17 5A 26 F8 20 9B DD 10 11 93 46 :5D
61C0 F9 27 F7 0D 4C 27 8E 17 0D BA 20 89 34 04 0D :35
61D0 26 34 86 01 97 32 86 00 C3 81 2B 26 29 1F 30 :17
61E0 04 54 C6 08 17 00 82 33 5B 17 04 76 17 03 A9 :C6

```

```

61F0 0B 17 00 99 5A 26 FA 17 03 BA 17 04 0C 7D 03 13 :90
Sum: E2 67 35 9B 8B EA 7B DD 18 BA 1E B1 05 7A B7 F8 :AF

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6200 10 26 FE 64 35 84 0D 4C 26 15 30 8D 0B 46 17 04 :4E
6210 55 17 03 F5 30 8D 0D 35 C6 11 6F 80 5A 26 FB 1F :C3
6220 30 17 04 12 C6 10 8D 41 86 3A 17 04 1C 96 4D 17 :F2
6230 04 08 0D 32 27 17 04 29 17 04 26 33 50 17 03 :A9
6240 57 C6 10 8D 48 5A 26 FB 17 03 39 17 03 8B 0C 4D :FD
6250 96 4C 81 10 26 A7 8D 4C 0F 4C 0D 31 27 06 17 03 :F9
6260 A8 17 03 45 17 03 A2 20 94 0F 4D 31 8D 0C DF 17 :F3
6270 03 F0 A6 C4 9D 4D 97 4D A6 C4 9B A4 A7 A0 A6 C0 :BF
6280 17 03 87 17 03 DC 5A 26 FB 17 03 39 17 03 8B 0C :FD
6290 31 27 0E 81 20 25 08 81 7F 27 04 81 FB 25 02 86 :85
62A0 2E 16 03 A5 0F 4D C6 39 86 2D 17 03 9C 5A 26 FB :28
62B0 30 8D 0A 97 17 03 AF 30 8D 0C 93 C6 10 A6 84 9B :60
62C0 4D 97 4D A6 80 17 03 72 17 03 97 5A 26 EF 86 3A :C3
62D0 17 03 76 96 4D 17 03 62 16 03 2E 17 01 87 27 06 :32
62E0 17 01 CC FD 03 14 BF 0F 14 1F 10 17 03 48 17 03 :7B
62F0 71 A6 B4 17 03 4A B6 2D 17 03 4E BD 31 4B 4B 4B :AA

```

Sum: C3 B3 31 49 BE 5E CB BE CE 20 CC C7 4A C0 96 14 :3A

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6300 4B 97 4C 8D 29 9A 46 A7 80 BF 03 14 1F 10 C4 0F :C6
6310 26 17 86 07 17 03 32 86 17 03 2D 30 10 C6 10 :33
6320 4F AB 80 5A 26 FB 17 03 11 17 02 DD 20 8B 17 02 :07
6330 47 1F 89 17 01 AB 25 07 34 02 17 03 07 35 B2 86 :72
6340 0A C1 2A 27 F2 C1 2F 27 23 C1 2B 27 1E C1 2D 27 :BF
6350 19 C1 2D 27 14 C1 2C 27 0F C1 1D 27 12 C1 1F 27 :93
6360 12 C1 1E 27 14 16 F0 4D 4C 4C 4C 4C 4C 4C 4C :1F
6370 1F 20 02 30 31 32 62 20 90 9F 44 30 1F 26 4C :62
6380 10 C4 F0 1F 01 B6 00 C3 B1 2B 27 E9 1F 13 17 02 :61
6390 7B 17 FE 3B 17 02 10 10 07 0B 27 07 1F 01 80 9E :64
63A0 44 20 DB 9E 44 BF 03 14 32 62 16 FF 39 1F 43 31 :69

```



## FM-7 EX-MOM ダンプ・リスト

63B0	BD	07	6C	04	34	04	BD	3A	A6	C4	17	02	7C	17	00	:DF	
63C0	B1	27	0A	17	00	E9	40	10	26	FC	DD	E7	C4	33	41	:92	
63D0	04	5A	26	E1	C6	04	34	04	BD	19	DD	C4	17	02	57	:44	
63E0	00	90	27	05	17	00	DB	ED	C4	33	42	35	04	5A	26	:E6	
63F0	16	FC	BA	A6	A0	C7	D2	51	AE	A0	17	02	4C	16	02	:62	
Sum:	7C	EA	A5	0B	60	BC	D6	E2	17	AF	41	D3	AD	12	35	:EE	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6400	86	7E	B7	01	D7	30	BD	FC	36	FF	01	DB	30	BD	08	:24	:06
6410	86	01	34	02	17	02	BD	17	02	4B	BD	86	31	C6	04	:76	
6420	86	24	17	02	24	5A	26	F8	BD	4B	27	17	17	00	80	:F0	
6430	01	1F	02	A6	A4	97	03	86	3F	A7	64	A1	64	19	26	:FC	
6440	67	A7	84	36	04	35	02	4C	81	05	26	06	16	FC	5E	:FC	
6450	01	17	01	E2	8D	10	27	FB	B1	24	26	05	6F	84	66	:07	
6460	47	98	01	20	FE	8D	48	74	26	A6	97	67	90	31	25	:06	
6470	20	BD	06	20	17	01	D2	17	01	9D	17	01	24	20	16	:D0	
6480	39	B7	02	6F	16	01	47	8D	9E	10	27	BD	1B	81	30	:10	
6490	26	FC	15	26	06	AE	9F	00	03	27	68	81	20	16	04	:06	
64A0	07	20	FE	79	7C	00	DA	28	63	11	00	19	25	BD	16	:06	
64B0	4A	0F	48	3B	0E	10	27	FB	EF	8D	26	10	25	1B	19	:06	
64C0	04	97	4E	06	9E	00	D9	8D	18	25	11	C6	04	00	40	:08	
64D0	4A	5A	26	26	E9	9A	4B	9D	4B	BD	C6	06	4C	36	F5	:04	
64E0	39	81	70	25	12	81	36	25	0A	81	41	25	00	81	42	:74	
64F0	05	BD	97	00	70	39	43	43	79	A6	BD	05	25	10	49	:4E	
Sum:	FA	AF	0D	9D	38	CC	02	F7	39	9B	14	0D	4A	1C	8D	:3F	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6500	4B	40	34	02	A6	80	BD	D9	35	00	AA	E4	72	81	29	:80	
6510	07	25	6E	11	82	8D	E1	1E	89	79	8D	97	18	FB	0E	:05	
6520	04	30	BD	1B	19	25	FB	89	1F	02	AE	80	31	26	1B	:06	
6530	91	30	19	26	FB	7D	8D	0C	25	00	90	90	31	01	04	:07	
6540	01	25	03	2B	76	86	04	34	24	19	39	AE	84	81	26	:26	
6550	04	74	01	20	F6	80	04	24	24	09	BD	97	25	E5	A7	:00	
6560	04	74	01	20	F6	80	04	24	24	09	BD	97	25	E5	A7	:00	
6570	01	25	03	2B	76	86	04	34	24	19	39	AE	84	81	26	:26	
6580	04	74	01	20	F6	80	04	24	24	09	BD	97	25	E5	A7	:00	
6590	04	74	01	20	F6	80	04	24	24	09	BD	97	25	E5	A7	:00	
65A0	04	74	01	20	F6	80	04	24	24	09	BD	97	25	E5	A7	:00	
65B0	04	74	01	20	F6	80	04	24	24	09	BD	97	25	E5	A7	:00	
65C0	04	74	01	20	F6	80	04	24	24	09	BD	97	25	E5	A7	:00	
65D0	04	74	01	20	F6	80	04	24	24	09	BD	97	25	E5	A7	:00	
65E0	05	A4	9E	04	9A	30	1F	00	A9	10	8F	00	4A	27	10	:10	
65F0	05	01	27	00	71	30	06	0A	D7	10	3F	26	F8	87	04	:04	
Sum:	1F	4C	20	55	FC	1C	3B	00	40	01	3D	1F	64	D8	54	:79	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6600	79	A8	84	81	01	27	0F	20	F5	86	9A	BD	30	FB	4A	:07	
6610	7B	BD	00	09	10	26	E9	BD	0A	70	34	16	86	11	F6	:01	
6620	05	FD	81	04	66	02	87	01	03	06	06	07	05	8E	07	:06	
6630	0A	02	81	7D	35	9A	BD	02	1F	9B	34	1E	0F	1F	9B	:06	
6640	8D	02	75	02	84	9E	17	03	04	1E	0F	1F	9B	34	1E	:06	
6650	08	DF	26	05	BD	D6	8E	35	9F	BD	02	75	94	74	10	:7E	
6660	D6	66	86	20	20	F6	80	04	24	09	BD	97	25	E5	A7	:00	
6670	77	17	00	01	20	00	08	BD	01	BD	84	86	2D	BD	09	:F7	
6680	90	29	27	06	8D	81	BD	06	20	0A	AE	81	BD	06	0D	:F7	
6690	BD	0D	0C	CF	6A	03	28	78	27	04	34	02	86	86	00	:9A	
66A0	35	02	9A	5A	26	F8	BD	19	DD	C4	33	42	35	04	5A	:26	
66B0	86	01	34	02	17	02	BD	17	02	4B	BD	86	31	C6	04	:76	
66C0	26	FD	01	04	27	17	02	BD	17	02	4B	BD	86	31	C6	:76	
66D0	01	04	27	05	BD	8C	5D	20	F7	8D	87	BD	05	A6	01	:34	
66E0	20	27	05	1F	FE	03	31	BD	06	3D	34	20	4F	E1	26	:0A	
66F0	17	FF	16	35	B7	3D	BD	06	07	06	20	C6	27	A7	80	:F8	
Sum:	05	BE	17	5D	D2	70	BD	19	AE	31	27	8B	A1	18	7C	:B2	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6700	26	FB	86	1F	97	28	BD	0E	36	72	4A	4B	20	BD	4A	:27	
6710	E7	86	1F	51	6C	8B	31	BD	33	AE	EF	A1	0F	E9	E6	:06	
6720	E7	86	1F	51	6C	8B	31	BD	33	AE	EF	A1	0F	E9	E6	:06	
6730	01	11	26	00	09	29	10	26	00	FF	00	29	20	EE	00	:29	
6740	30	BD	0A	E4	7A	66	0A	C1	B9	10	27	00	AE	00	29	:10	
6750	27	00	EE	C1	5A	26	1A	D6	03	54	10	24	00	95	30	:BD	
6760	04	C6	06	81	7A	0A	09	29	26	03	86	4C	8C	B6	55	:20	
6770	68	C1	57	26	00	0A	29	26	03	86	32	8C	B6	33	C6	:03	
6780	20	57	96	03	84	F0	10	20	26	19	0A	29	10	26	00	:A6	
6790	96	03	84	F0	10	20	00	9E	CC	4C	42	DD	07	C6	0A	:95	
67A0	20	20	36	03	81	80	25	4A	96	03	84	0F	81	0E	24	:5E	
67B0	16	81	0C	26	3E	E6	84	C1	18	26	38	0A	29	26	03	:86	
67C0	59	8C	86	53	5F	20	12	96	03	84	F0	10	20	03	86	:4A	
67D0	59	8C	86	53	5F	20	12	96	03	84	F0	10	20	03	86	:4A	
67E0	E6	84	30	BD	03	87	3A	31	BD	07	1E	AE	80	2B	06	:A7	
67F0	A0	20	F8	20	43	84	7F	A7	A0	0D	27	26	29	0D	29	:44	
Sum:	A1	FD	2B	DB	E6	B9	44	05	4F	66	DB	D2	E2	BD	0B	:5E	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6800	04	96	02	20	02	96	03	84	F0	26	0A	86	06	20	15	:44	
6810	44	44	44	44	44	44	44	44	44	44	44	44	44	44	44	:44	
6820	44	44	44	44	44	44	44	44	44	44	44	44	44	44	44	:44	
6830	08	8C	86	07	20	EE	00	29	C6	0B	D7	27	4F	C6	8B	:20	
6840	A1	D6	02	C1	80	25	48	C1	80	25	03	86	42	8C	86	:41	
6850	04	0F	C1	03	26	04	86	44	20	31	C1	0C	25	2D	C1	:09	
6860	26	12	E6	84	C1	64	27	EE	C1	91	27	02	5F	8C	C6	:09	
6870	86	20	18	D6	02	C1	C0	24	04	86	56	5F	10	0D	D6	:02	
6880	04	0F	C1	0E	24	03	86	44	8C	B6	55	1F	16	FF	4C	:7E	
6890	F0	C1	60	24	0F	C1	40	25	0E	26	03	86	41	8C	86	:42	
68A0	C6	03	20	E8	5F	20	C9	C1	10	24	02	20	C3	C1	20	:26	
68B0	06	86	42	97	08	20	B7	C1	20	26	12	D6	02	C4	0F	:C1	
68C0	06	24	04	C6	08	20	A9	C1	08	24	04	C6	01	20	A1	:C1	
68D0	0C	26	03	C6	04	8C	0C	86	20	20	B0	D6	02	C4	0F	:75	
68E0	01	0E	25	04	C6	02	20	8B	C1	0D	27	EA	C1	0A	25	:06	

68F0	C6	04	86	43	20	96	C1	09	27	DC	C1	06	25	D8	C6	:0A	
Sum:	BA	59	CB	DB	85	AD	A4	A7	10	3C	C0	1C	57	99	09	:E3	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6900	20	D6	C6	00	D7	04	0F	2C	74	04	54	54	54	54	54	:F8	
6910	05	FC	31	21	8D	19	C6	2C	67	04	35	04	30	01	C4	:0F	
6920	BD	0D	04	20	20	01	29	0C	7A	2A	DD	0F					

## FM-7 EX-MOM ダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6E00	6E	6B	79	20	53	6F	66	74	20	44	65	73	69	67	6E	65	:ED
6E10	64	20	42	79	20	54	2E	4B	61	73	61	73	61	6B	75	2E	:43
Sum:	D2	8B	B8	99	73	C3	94	BF	B1	B7	C6	E6	CA	D2	E3	93	:30

## FM-7 EX-BASIC ダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6B00	35	10	BF	03	16	30	BC	F6	9C	45	24	11	DC	45	93	3F	:DB
6B10	34	06	9F	45	1F	10	A3	E1	DD	3F	BD	8F	51	86	09	B7	:D0
6B20	02	03	30	BD	04	37	BF	02	04	30	BD	00	0E	BF	02	10	:5E
6B30	30	BD	04	57	BD	9B	DB	6E	9F	03	16	81	F4	27	51	81	:DF
6B40	F5	10	27	02	EE	B1	F6	10	27	01	73	7E	92	A0	11	00	:FF
6B50	04	48	00	80	00	80	00	00	1B	00	00	00	00	00	00	00	:67
6B60	00	07	01	02	03	04	05	06	07	9D	D2	8D	92	8F	BD	99	:C6
6B70	F9	34	04	BD	92	BC	35	04	C1	19	25	03	7E	96	63	9E	:5C
6B80	4B	8C	00	50	24	F6	39	00	00	00	00	00	00	00	00	00	:7A
6B90	BD	D7	E7	8C	F7	1F	10	E7	BC	F1	9D	DB	81	DA	27	03	:5B
6BA0	7E	92	A0	BD	C4	E7	BC	E6	1F	10	E7	BC	E0	6F	BC	D9	:B0
6BB0	6F	BC	D7	9D	D8	27	14	81	2C	26	E5	9D	D2	BD	9A	02	:02
6BC0	BC	02	80	25	03	7E	96	63	AF	BC	BE	A6	BC	BD	A1	BC	:C2
6BD0	BC	24	0D	34	02	A6	BC	B5	A7	BC	B0	35	02	A7	BC	AD	:04
6BE0	A6	BC	A9	A1	BC	AB	24	0D	34	02	A6	BC	A1	A7	BC	9C	:B9
6BF0	35	02	A7	8C	99	A6	BC	93	A0	C2	92	AC	C6	08	3D	E3	:C0
Sum:	75	6E	F9	F9	5A	38	B4	67	27	3B	FD	13	F9	8F	63	54	:33

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6900	BC	B7	ED	8C	B2	A6	BC	B5	A7	BC	B4	B6	1B	8D	60	B6	:90
6910	4C	BD	5C	A6	BD	FF	71	BD	56	A6	BD	FF	6A	BD	50	10	:44
6920	AE	BD	FF	65	27	07	4F	BD	46	31	3F	26	F9	BD	45	A6	:F6
6930	BD	FF	5B	A1	BD	FF	56	27	06	6C	BD	FF	50	20	EE	A6	:90
6940	BD	FF	4C	A7	BD	FF	46	A6	BD	FF	41	A1	BD	FF	3F	27	:57
6950	14	6C	BD	FF	39	B6	1B	BD	16	B6	41	BD	12	86	08	BD	:0A
6960	0E	BD	0A	20	A6	B6	1B	BD	06	B6	32	BD	02	B6	0A	3A	:AA
6970	10	7E	D6	66	BE	04	04	34	10	31	BD	FE	D1	C6	1B	A6	:F4
6980	A0	A7	BD	5A	26	F9	35	10	A6	BD	FF	01	C6	08	3D	ED	:B0
6990	0B	C3	00	07	ED	0F	A6	BD	FE	F4	C6	08	3D	ED	0D	C3	:BE
69A0	00	07	ED	8B	11	9D	DE	BE	04	4C	C6	0B	34	14	C6	0B	:CA
69B0	68	BD	49	5A	26	FA	BD	B7	35	14	5A	26	EF	39	9D	D2	:4F
69C0	BD	E0	5B	9E	C5	D6	C8	BF	05	04	F7	05	06	BD	99	FD	:16
69D0	F7	05	0A	BD	99	FD	F7	05	0B	B6	05	0A	3D	4D	26	19	:EE
69E0	B6	20	3D	27	14	BD	BD	AC	B6	05	0B	48	48	48	48	BB	:B5
69F0	05	06	25	05	4A	B1	C8	25	05	C6	05	7E	BD	D1	B7	05	:55
Sum:	24	12	D6	2E	C3	6A	BB	31	AA	71	0F	6F	7E	FD	BA	D0	:EE

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6A00	07	B6	01	E5	97	F8	0F	C2	9D	DB	B1	3B	27	2F	BD	92	:DC
6A10	92	B1	2C	27	12	BD	99	BC	C1	0B	25	03	7E	96	63	D7	:C9
6A20	FB	9D	DB	B1	3B	27	16	BD	92	92	BD	E2	24	03	7E	16	:C6
6A30	92	A0	D7	C2	9D	D2	B1	3B	27	03	7E	92	A0	9D	D2	26	:65
6A40	01	39	B6	05	0A	C6	10	3D	FE	05	04	DF	C5	33	CB	11	:CC
6A50	B3	02	80	22	35	FF	05	04	33	5F	DF	C9	FC	05	06	0F	:B4
6A60	C7	0F	CB	97	CB	D7	CC	BD	92	C3	BD	9A	05	DE	3D	BD	:B9
6A70	25	B6	FF	97	F8	7F	01	A5	DF	F5	DE	3D	DF	F3	BD	EA	:C6

6A80	4A	9D	DB	27	0D	BD	92	92	20	B8	9D	DB	27	04	9D	D2	:B8
6A90	26	FC	39	FD	05	0A	34	08	B6	05	1F	BB	1F	10	BE	05	:9A
6AA0	A0	84	7F	C4	7F	ED	04	B6	16	A7	B4	DC	0A	3D	10	B3	:54
6AB0	00	01	23	5F	CC	05	10	ED	02	BD	F1	7D	1F	01	10	BE	:3C
6AC0	00	0B	96	0B	97	09	B6	10	97	0C	A6	B4	97	0B	96	0A	:EB
6AD0	0B	08	1F	A9	20	02	1F	9A	69	C4	31	3F	26	06	33	41	:F0
6AE0	10	BE	00	0B	4A	26	EF	0A	0C	D6	0C	27	0B	C1	08	26	:1B
6AF0	DD	30	01	20	D5	0A	09	27	04	30	1F	20	C9	30	01	BC	:36

Sum: 9B 30 45 C7 B3 C0 9B 01 B7 8B 92 F3 C9 E0 DD 89 :86

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6B00	05	30	25	BE	1F	20	C1	0B	27	05	68	C4	5A	26	F8	33	:26
6B10	41	35	8B	EF	02	BD	F1	7D	33	CB	14	35	8B	00	00	00	:E6
6B20	6C	D1	4F	A7	8C	F7	9D	D2	9D	D8	27	56	B1	2C	27	03	:EE
6B30	7E	92	A0	9D	D2	B1	23	27	26	B1	5D	27	E5	B1	5B	27	:FD
6B40	E2	B1	DC	27	0B	B1	40	27	11	BD	9A	02	BD	15	20	DB	:5D
6B50	BD	31	6F	BC	C8	6F	BC	C6	0C	CC	A7	BC	C1	20	C7	BD	:96
6B60	73	20	EF	EE	BC	BA	AF	C1	A6	BC	B2	A7	C0	EF	BC	BD	:9C
6B70	4D	27	03	6C	8C	A9	6C	8C	A6	A6	BC	A3	B1	2B	25	02	:5B
6B80	BD	01	39	6D	8C	99	26	04	B6	0A	20	55	17	00	B2	CE	:1F
6B90	04	40	BD	50	B6	1B	BD	49	B6	4A	BD	45	B6	01	6D	BD	:BB
6BA0	FF	7C	27	02	B8	17	BD	39	B6	0D	BD	35	17	00	92	CE	:DB
6BB0	04	42	6D	BD	FF	6B	27	02	33	4E	BD	28	B6	1B	BD	21	:55
6BC0	B6	4A	BD	1D	B6	23	6D	BD	FF	54	27	02	B8	0D	BD	11	:CF
6BD0	B6	0D	BD	0D	30	BD	00	F9	AF	BD	FF	44	6F	BD	FF	3F	:9C
6BE0	39	16	FD	B8	31	BD	00	E9	E6	BD	FF	33	34	44	EC	A1	:2B
6BF0	ED	BD	00	AD	30	BD	00	A5	9D	DE	34	40	BD	1F	35	40	:99

Sum: 25 BA 4A AC 1D A5 2D 54 90 DC 9F FE CC 3B 00 EF :14

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6C00	33	41	BD	19	C6	02	4F	BD	DB	6D	A4	27	02	BD	D2	5A	:B9
6C10	26	F4	35	44	6D	A0	27	01	5A	5A	26	D0	39	C6	0B	34	:AD
6C20	44	C6	0B	68	C4	A9	6D	BD	FE	F4	26	02	33	42	33	42	:85
6C30	5A	26	F0	BD	AC	6D	A4	27	02	BD	A6	35	44	5A	26	DF	:EE
6C40	39	4F	BD	9D	B6	1B	BD	99	B6	4C	BD	95	B6	12	E6	BD	:78
6C50	FE	CD	3D	34	02	1F	9B	BD	B8	35	02	20	B4	43	4B	A1	:B1
6C60	49	CE	45	52	41	53	C5	4C	4C	49	53	D4	4C	50	52	49	:46
6C70	4E	D4	53	4F	55	4E	C4	50	4C	41	D9	43	4F	50	D9	4B	:E7
6C80	50	52	49	4E	D4	53	59	4D	42	4F	CC	45	7B	70	61	3C	:3C
6C90	6E	64	65	64	20	46	2D	42	41	53	49	43	00	16	00	04	:AA
6CA0	40	00	00	45	5B	50	41	4E	44	45	52	20	56	65	72	2E	:12
6CB0	32	2E	30	3B	33	F2	33	F3	37	F4	20	28	43	29	20	42	:54
6CC0	59	20	45	54	46	20	4C	61	62	6F	72	61	74	6F	72	79	:97

Sum: 4E E3 3F 47 B6 26 75 41 43 90 CD B2 A9 6F 00 5F :E2



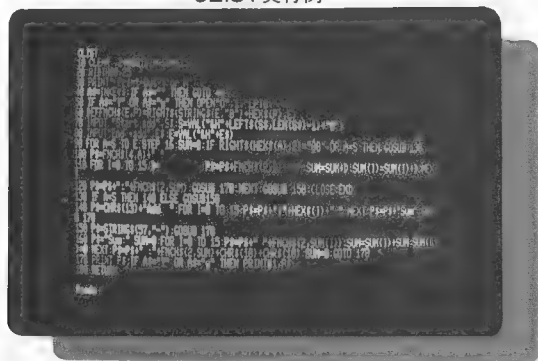
FM-7/8

BASICプログラムの開発,  
デバッグ, 解析に!

# BASIC ANALYZER

■平井秀明

CLIST実行例



「BASIC ANALYZER」はBASICプログラムの開発、デバッグ、解析を短時間に少ない労力で行なうためのツールで、F-BASICにCLIST, VLIST, FIND, SYSTEMの4つのコマンドを新たに拡張します。これらのコマンドは一般のコマンドと同じように、ダイレクトで、またはプログラム中でも使用できます。また、すべてのコマンドでプリンタへの出力が可能であり、特にCLISTコマンドではカラープリンタへの出力もサポートされています。

プログラム自体は2.5Kバイトと小さく、ポジション・インディペンデントなので、メモリ上の好きな所に配置でき、20Kバイト以上の大きなプログラムの開発にも使えます。

## 初期設定

本プログラムは完全にリロケータブルですが、リストどおりの位置(\$7600~\$7FFF)にあるものとして説明します。まず、CLEAR、&H75FF、としてエリアを確保してください。カラーブロック・プリンタを使う方はその機種によって、次のようにプログラムを書き換えてください。

- ① タンディ製CGP-115  
POKE &H7602,0
- ② NEC製PC-6022  
POKE &H7602,1
- ③ 精工舎製GP-700M  
POKE &H7602,2

テキスト・モードで色指定のコントロール・コードが

CHR\$(29)で次の色に変わるタイプが①、CHR\$(27)+CHR\$(97~100)で4色を指定するタイプが②、CHR\$(20)+CHR\$(48~54)で7色を指定するタイプが③になります。

他の機種の場合でも、上記のどれかとコントロール・コードが一致すれば使えると思います。なお、GP-700Mについては発売前に直接メーカーに問い合わせたものなので、仕様変更などがあると使えない場合があります。

FM-7と8で、またはBASICのバージョンによって次のように書き換えてください。

- ① FM-8の場合  
POKE &H7601,8
- ② F-BASICのバージョンが1の場合  
POKE &H7DD1,&HED  
POKE &H7DCD,&HB3
- ③ F-BASICのバージョンが2の場合  
POKE &H7DD1,&HF1

## コマンドの説明

### 〔CLIST〕

#### 説明

CLISTはBASICプログラム中の命令をその働きに応じて色分けして出力するものです。

CLISTの出力を見ると一見して全体の構造がわかります。そもそも単色のリストが見にくい原因は、本来立体的構造を持つプログラムというものを平面に、あるいは直線的に表わしているところにあります。これがCLISTによって立体的に表現されます。以て例として地図があります。質の良い多色刷りの地図は、1色刷りの地図よりはるかに見やすく、一見して川と道と境界線の区別や高低の区別がわかります。

なお、CLISTには画面でリストを追いつながら文字列検索をする機能もついています。

CLISTで表示されたカラーリストはそのまま普通のリスト同様にエディットすることができます。

色分けは表1のようになっています(画面出力の場合)。FM-7ではパレット命令で表示色を変えることも可能です。

#### 使い方

F-BASICのLISTに準じています。特別な使い方として次のようなものがあります。

図1 VLIST表示例

変数/属性	10進の値	10進の値/キャラクタ	格納アドレス
整数型	65	(41/"A")	[0828]
	13	(0D/"."	[082C]
{ P#	"B"	(42)	[0830]
{ X#	"ABCD."	(41,42,43,44,0A)	[0835]
		ASCIIコード	
配列	A!(2,1)		
	(0)	12!	[0841]
	(1)	1546!	[0845]
	(2)	23!	[0849]
	(3)	452!	[084D]

表1 CLISTカラー

分	色
注意書き(REM文など)	白色反転
エディタ・テバッグ(MONなど)	紫色反転
カセット、プログラムの入出力(SAVEなど)	青色反転
音声(BEEP,PLAYなど)	緑色反転
初期設定(DEFなど)	紫色
マシン語の取り扱い(POKE,EXECなど)	赤色
データ入力(INPUT,READなど)	青色
画面表示(PRINT,LINEなど)	緑色
プログラムの流れに関するもの(GOSUBなど)	水色
変数処理、代入、計算(LETなど)	白色
ファイルへの出力、ディスク関係	"
拡張命令	"
行番号、セパレータ	黄色

注) 2通り以上の働きのある命令は働きによって色を変えます。

表2 ROM内ルーチン

機能	FM-7	FM-8
エラー(IN条件: Bレジスタにエラーコード)	8DD1	92AB
コマンド・レベルへ	8E72	9345
文字列を読み取り文字領域へ(OUT条件: Xレジスタにアドレス, Bレジスタに文字数)	98F1	9D88
復改, 改行	9B50	9F97
Xレジスタから00を見つけるまで出力	D678	DB1B
Xレジスタ1から00を見つけるまで出力	9BDB	9FDF
空白1つ出力	9C22	A026
Dレジスタの内容を16進出力	AC37	AFE3
Aレジスタ "	AC3D	AFE9
Dレジスタの内容を10進出力	B615	B96D
CLOSE	CE4B	D10F
OPEN	CEDF	D1A3
1文字入力(IN条件: Aレジスタ)	D072	D336
1文字出力(OUT条件: Aレジスタ)	D08E	D352
ABORTテスト	D678	D8C2
BREAKテスト	DB59	DD58
CLS	DA80	DC89
BIOS	ABFA	ABFA
BEEP(ON=\$81, OFF=\$00)	FD03	FD03
16進で表わされる定数を文字列にコンバート(OUT条件: Xレジスタに文字列先頭アドレス, IN条件: \$17に属性(1,2,3,4,8)をセットし\$76に16進数をセット, 属性4,8のときはFM-7は\$B301, FM-8は\$B676をコール)	B620	B978
テキスト中で\$FEF2~の形で表わされる定数をコンバートするのに先立ってコールする。	974D	9BEF
	BD19	C083

注) 属性1=1バイト整数, 2=2バイト整数, 3=文字型, 4=単精度型, 8=倍精度型。

## CLIST[n-m] "文字列"

nmは行番号で省略可。

"."は"."でも可。

この命令では指定された文字列を捜しながらn行からm行まで表示します。

また、LISTと違ってマルチステートメントが可能なので、CLIST10:CLIST50:CLIST100-200のような使い方ができます。カラープリンタへの出力はCLIST@, CLIST@100のようにします。

## [VLIST]

## 説明

これは変数のリストを出力するもので、カシオのC82-BASICのVARLIST命令を拡張した機能を持っています。

VLISTを実行すると、その時点で登録されている変数名をすべて表示します。必要に応じて整数型だけを表示したり、文字型と倍精度型だけを表示することができます。また、変数の内容や格納されているアドレスの一覧表も出力します。この場合、さらに整数型はその16進表示、キャラクタ表示を行ない、文字型はASCIIコードの表示を行ないます。

表示例を図1に示します。配列では必要以上の表示を避けるため、値が0や""(ヌルストリング)のときは何も表示しません。ただし、すべての要素が0や""でも配列名は表示します。また、配列番号はすべて通し番号で表示されます。DIM文でDIM S(a,b,c)と宣言された配列でS(i,j,k)の配列番号nは

$$n = i + j \times (a+1) + k \times (a+1) \times (b+1)$$

となります。たとえば、DIM S(1,2,3)と宣言した配列のS(1,2,3)が10なら $1+2 \times 2+3 \times 2 \times 3=23$ となるので、

(23) 10 [08A8]

のように表示されます。

## 使い方

VLISTを単に実行すると変数名のみをすべて出力します。VLISTには%, !, #, \$, \*, /の6つのオプションがあり、それぞれのような意味があります。

%: 整数型変数の出力

!: 単精度型の出力

#: 倍精度型の出力

\$: 文字型の出力

\*: これを付けると変数名の他にその内容、格納アドレ

スなどを出力します。

/: これを付けた場合、\*がないときは単に変数ごとの改行を行ない、\*があるときは配列の要素数が25を越える配列の内容を省略して、変数名だけを表示します。これは事務系プログラムなどで、非常に大きい配列が使われているとき、表示を省略するためのものです。

## 例

VLIST # \$ \* 倍精度型、文字型をその内容と共に出力します。

VLIST % / 整数型変数名のみを改行しながら出力します。

VLIST @ # ! \$ \* 全部の変数名をプリンタに出力します。

なお、VLISTではリスト上には存在しても実行時に使われない変数は表示されません。

[FIND]

## 說明

これは指定した文字列を含む行のみを表示するコマンドです。たとえば、“HSS”という変数名をすべて“HISCORE”と変えたい場合は、このコマンドで“HSC”を含む行だけを表示させて、修正するといった使い方ができます。

このFIND機能はCLIST機能と共に、BASIC テキストの内部表現に左右されません。たとえば“NTA\$”のような指定で“PRINT A\$”を捜すことができます。

## 使い方

CLISTとまったく同じですが、文字列が見つかった行のみを表示します。出力は単色です。

<b>例</b>	
<b>FIND 10,50 "ABC"</b>	10行から50行の間で、ABCを含む行のみ出力します。
<b>FIND @,-100 "DEF"</b>	直前に表示した行から100行の間でDEFを含む行のみ、プリンタへ出力します。

表3 BASICワーク(FM-7, FM-8 共通)

機 能	アドレス
テキスト・ポインタ	0033
変数開始 (DEFFN含む) ポインタ	0035
配列開始ポインタ	003B
フリーエリア・ポインタ	003D
リスト・スタート・アドレス	00A2
リスト・エンド・アドレス	00A4
現在実行している行	00A6
ファイル・ナンバー	00BF
デバイス・ナンバー	02E6
WIDTH	00C3
バッファから1文字読む	00D8
バッファから1文字読みポインタを1つ進める	00D2
色のコード	01E5
ステートメント・テーブル	01EF
バッファ・トップ・アドレス	043D
バッファ・エンド・アドレス	053C
スタック下限ポインタ	003F
文字領域下限ポインタ	0045

## [SYSTEM]

## 說明

これも C82-BASICのSYSTEM コマンドにヒントを得たもので、BASIC プログラムのテキスト・サイズ (10進数で表示され、1024で割ると何Kバイトかわかる) の他、BASIC プログラム・テキストの格納先頭アドレス、ユーザー定義関数の格納先頭アドレス (=BASIC プログラム・テキストの格納最終アドレス + 1)、フリーエリア先頭アドレス、システム・スタック・エリアの最終アドレス、BASIC エリアの最終アドレスを表示します。

ちなみに**CLEAR 300,&H6FFF**を実行すると、**BASIC**エリア最終アドレス=&H6FFF、システム・スタック・エリア最終アドレス=&H6FFF-300となるので、**CLEAR**文がどこに設定されているかわからなくなったときにも使えます。他のコマンドと同様にプリンタ出力のときは“@”をつけてください。

## プログラムについて

20種程度のROM内ルーチンと20種程度のBASICエリアを参照しています。したがってFM-7使用時には、ロードするとき自動的にFM-7用にジャンプ・テーブルを書き換えています。使用したジャンプ・テーブルを表2に、BASICワークを表3に示します。

最も苦勞した部分は、一文字出力と色指定を普通にB I O Sコールして行なうとフィールドが設定されるため、C LIST表示の後エディットができないことでした。あの「Y AMAUCHI」を使って直接サPCUTのアドリビュート・パッファ書き換えを試みたのが不発に終わりました。

“PUT CHARACTERLINE”のコマンドを使って、カラー出力したところ、フィールドが設定されないことがわかりうまくいきました。

蛇足かもしれませんが、このプログラムをロードした後（もちろん、オート・スタートで）先頭から\$90バイトは不要なので、BASICプログラムが大きくなったときは、**CLEAR**文を設定し直してください。

## ■BASIC ANALYZERアセンブル・リスト

```

1      *
2      *| BASIC ANALYZER V2.0 for FM-7/8
3      *|
4      *|   file name "ANAL7600"
5      *|
6      *|   by H.HIRAI 83/4/21-23:10
7      *|
8      *|
9
10 7600      ORG      $7600      START ADD.
11
12      *
13      * BASIC WORK
14      *
15 0017      cstype EQU   $17
16 0033      text EQU    $33
17 0035      simple EQU  $35
18 0038      array EQU   $38
19 003D      free EQU    $3D
20 0076      csset EQU   $76
21 00A2      start EQU  $A2
22 00A4      end EQU     $A4
23 00A6      curent EQU  $A6
24 00BF      fileno EQU  $BF
25 00C3      width EQU   $C3
26 00D2      getinc EQU  $D2
27 00D8      get EQU     $D8
28 01E5      color EQU   $1E5
29 01EF      table EQU   $1EF
30 02E5      device EQU  $2E5
31 043D      bufstop EQU $43D
32 053C      bufend EQU  $53C
33 003F      stack EQU   $3F
34 0041      string EQU  $41
35 0045      basic EQU    $45
36      *
37      * CONSTANT
38
39
40
41
42 7600 8C      LINK  FCB  CMPX,7,1
43 7603      C6 39      LDB  #539
44 7605      E7 8CF8     STB  LINK,PCR
45 7608      EC 8CF6     LDD  <LINK+1,PCR
46 7608      5A          DECB
47 760C      27 06       BEQ  LINK0
48 760E      5A          DECB
49 760F      27 05       BEQ  LINK1
50 7611      C6 22       LDB  #CPR11-SW3P-1
51 7613 8C      FCB  CMPX
52 7614      C6 15       LINK0 LDB  #CPR12-SW3P-1
53 7616      E7 8D6533  LINK1 STB  SW3P,PCR
54 761A      81 07       CMPA  #7
55 761C      26 12       BNE  LINK3
56 761E      85 14       LDA  #29
57 7620      38 8D89B4   LEAX  JTABLE,PCR
58 7624      31 8C7A     LEAY  <ERROR,PCR
59 7627      EE 81       LINK2 LDU  ,X++
60 7629      31 21       LEAY  ,Y++
61 762B      EF A1       STU   ,Y++
62 762D      4A          DECA
63 762E      26 F7       BNE  LINK2
64 7630      CC 0015     LINK3 LDD  #8015
65 7633      CE 01EF     LDU   #table
66 7636      AB CA       LINK4 ADDA ,U
67 7638      33 4A       LEAU  10,U
68 763A      6D CA       TST   ,U
69 763C      26 F8       BNE  LINK4
70 763E      81 FA       CMA   #5FA
71 7640 1022 0187      LDBH  ERR21
72 7644      AE 43       LDX   3,U
73 7646      34 12       PSHS  A,X
74 7648      17 0330     LBSH  NAME

```

## BASIC ANALYZERアセンブル・リスト

```

75 7648 30 8D008E LEAX START,PCR
76 764F AF 43 STX 3,U
77 7651 1183 020D CMPL #table+30
78 7655 27 07 BEQ LINK5
79 7657 1022 0170 LBHI ERR21
80 7658 AF 4D STX 13,U
81 765D 8C FCB #10,U
82 765E 33 56 LINK5 LEAU -10,U
83 7660 AE 41 LDX 1,U
84 7662 31 8D0915 LEAY NAME,PCR
85 7666 10AF 41 STY 1,U
86 7669 AE C4 LDA ,U
87 766B 8D 1D BSR LINK7
88 766D 30 8C1D LEAX <NAME,PCR
89 7670 86 04 LDA #4
90 7672 8D 16 BSR LINK7
91 7674 35 12 PULS A,X
92 7676 A7 8C1A STA <code,PCR
93 7679 AF 8C18 STX <other,PCR
94 767C AE C4 LDA ,U
95 767E 8B 04 ADDA #4
96 7680 A7 C4 STA ,U
97 7682 39 RTS
98
99 7683 E6 80 *-----
100 7685 E7 A0 LINK6 LDB ,X+ Trans NAME
101 7687 2A 01 STB ,Y+
102 7689 4A BPL LINK7
103 768A 26 F7 LINK7 BNE LINK6
104 768C 39 RTS
105
106 * NEW STATEMENT TABLE
107
108 768D 46 NAMES FCC /FIN/
109 7690 C4 FCC #C4 D
110 7691 43 FCC /CLIS/
111 7695 D4 FCB #D4 T
112 7696 53 FCC /SYSTE/
113 7698 CD FCB #CD M
114 769C 56 FCC /ULIS/
115 76A0 D4 FCB #D4 T
116
117 * FM-8 JUMP TABLE
118
119 76A1 7E 92A8 ERROR JMP #92A8
120 76A4 7E 9D86 STRING JMP #9D86
121 76A7 7E 9F97 CR JMP #9F97
122 76AA 7E 9FDF WRITE2 JMP #9FDF
123 76AD 7E A026 SPACE JMP #A026
124 76B0 7E AFE3 HEXD JMP #AFE3
125 76B3 7E AFE9 HEXA JMP #AFE9
126 76B6 7E B676 OV4SET JMP #B676 SET SNG.
127 76B9 7E B96D PRINTD JMP #B96D
128 76BC 7E B978 CONN JMP #B978 TO ASCII
129 76BF 8D 9BEF F2SET1 JSR #9BEF
130 76C2 7E C083 F2SET2 JMP #C083
131 76C5 7E D10F CLOSE JMP #D10F
132 76C8 7E D1A3 OPEN JMP #D1A3
133 76CB 7E D335 INPUT JMP #D335
134 76CE 7E D352 PRIA JMP #D352
135 76D1 7E D81B WRITE JMP #D81B
136 76D4 7E DC89 CLS JMP #DC89
137 76D7 8D DD58 BREAK JSR #DD58
138 76DA 7E D8C2 ABORT JMP #D8C2
139
140 FBFA BIOS EQU #FBFA
141 FD03 BEL EQU #FD03 on=#01,off=#
142
143 * PROGRAM WORK
144
145 7692 WORK EQU ERROR-15
146 7693 code EQU WORK+1 #F4etc.
147 7694 other EQU WORK+2 (2byte)
148 7695 strno EQU WORK+3
149 7697 stradd EQU WORK+4 (2byte)
150 7699 flag1 EQU WORK+5 DATA,REM,', "
151 769A flag2 EQU WORK+6 .
152 769B flag3 EQU WORK+7 EorD
153 769C flag4 EQU WORK+8 FIND
154 7699 vflag EQU flag1 /= *%!*%
155 769D SW1 EQU WORK+11
156 769E SW2 EQU WORK+12
157 769F SW3 EQU WORK+13
158 76A0 SW31 EQU WORK+14
159
160 * STATEMENT START
161
162 76DD 4F START CLRA
163 76DE 5F CLRB
164 76DF ED 8C8B STD SW1,PCR
165 76E2 ED 8CBA STD SW3,PCR
166 76E5 9D D8 JSR get
167 76E7 A0 8CA9 SUBA code,PCR
168 76EA 1027 05C1 LBEQ FIND
169 76EE 4A DECA
170 76EF 1027 #5C0 LBEQ CLIST
171 76F3 4A DECA
172 76F4 27 0A BEQ SYS
173 76F6 4A DECA
174 76F7 1027 0102 LBEQ VLIST
175 76F8 9D D8 JSR get
176 76FD 6E 9C94 JMP [other,PCR]
177
178 * SYSTEM
179
180 7700 9D D2 SYS JSR getinc
181 7702 81 40 CMPA #0
182 7704 26 03 BNE SYS0
183 7706 17 00D1 LBSR LOPEN
184 7709 30 8C37 SYS0 LEAX <ST0-1,PCR

```

```

185 770C 8D 9C BSR WRITE2
186 770E DC 35 LDD simple
187 7710 93 33 SUBD text
188 7712 8D A5 BSR PRINTD
189 7714 8D 91 BSR CR
190 7716 30 8C41 LEAX <ST1-1,PCR
191 7719 DC 33 LDD text
192 771B 8D 1A BSR SYSOUT
193 771D 30 8C50 LEAX <ST2-1,PCR
194 7720 DC 35 LDD simple
195 7722 8D 13 BSR SYSOUT
196 7724 30 8C5F LEAX <ST4-1,PCR
197 7727 DC 3D LDD free
198 7729 8D 0C BSR SYSOUT
199 772B 30 8C6E LEAX <ST5-1,PCR
200 772E DC 3F LDD stack
201 7730 8D 05 BSR SYSOUT
202 7732 30 8C7D LEAX <ST7-1,PCR
203 7735 DC 45 LDD basic
204 7737 34 06 SYSOUT PSHS A,B
205 7739 17 FF6E LBSR WRITE2
206 773C 35 06 PULS A,B
207 773E 17 FF6F LBSR HEXD
208 7741 16 FF63 LBRA CR
209
210 7744 0D ST0 FCB 13
211 7745 42 FCC /BASIC PROGRAM SIZE = /
212 775A 00 FCC #
213 775B 54 ST1 FCC /TEXT TOP ADDRESS = /
214 7770 00 FCC #
215 7771 57 ST2 FCC /WORK TOP ADDRESS = /
216 7786 00 FCC #
217 7787 46 ST4 FCC /FREE TOP ADDRESS = /
218 7790 00 FCC #
219 779D 53 ST5 FCC /STACK END ADDRESS = /
220 77B2 00 FCC #
221 77B3 42 ST7 FCC /BASIC END ADDRESS = /
222 77C8 00 FCC #
223
224 * ERROR
225
226 77C9 C6 02 ERR2 LDB #2
227 77CB 16 FED3 ERR21 LBRA ERROR
228
229 * TAB
230
231 77CE 34 02 TAB PSHS A
232 77D0 17 FED4 LBSR SPACE
233 77D3 96 09 LDA #9
234 77D5 17 FF6F LBSR PRIA
235 77D8 35 82 PULS A,PC
236
237 * PRINTER OPEN
238
239 77DA 9D D2 LOPEN JSR getinc
240 77DC C6 11 LOPEN0 LDB #11
241 77DE D7 BF STB fileno
242 77E0 CC 0300 LDD #0300
243 77E3 FD 02E6 FD device
244 77E6 16 FEDF LBRA OPEN
245
246 * PRINT Accd in HEX
247
248 77E9 34 02 PRDH PSHS A
249 77EB 4D TSTA
250 77EC 27 08 BEQ PRAH0
251 77EE 17 FEBF LBSR HEXD
252 77F1 35 82 PULS A,PC
253
254 * PRINT AccA in HEX
255
256 77F3 34 02 PRAH PSHS A
257 77F5 BC FCB CMPX
258 77F6 1F 98 PRAH0 TFR B,A
259 77F8 17 FEB8 LBSR HEXA
260 77FB 35 82 PULS A,PC
261
262 * VLIST
263
264 77FD 6C 0DFE9C VLIST INC SW1,PCR
265 7801 6F 0DFE94 VLIST CLR vflag,PCR
266 7805 9D D2 JSR getinc
267 7807 81 40 CMPA #0
268 7809 26 0E BNE VLIST2
269 780B 8D CF BSR LOPEN0
270 780D 20 08 LBSR VLIST1
271 780F EA 0DFE96 VLIST0 ORB vflag,PCR
272 7813 E7 0DFE92 VLIST1 STB vflag,PCR
273 7817 9D D2 VLIST1 JSR getinc
274 7819 C6 20 VLIST2 LDB #20
275 781B 81 DC CMPA #0C
276 781D 27 F0 BEQ VLIST0
277 781F 54 LSRB
278 7820 81 DB CMPA #0B
279 7822 27 EB BEQ VLIST0
280 7824 54 LSRB
281 7825 81 23 CMPA #'
282 7827 27 E6 BEQ VLIST0
283 7829 54 LSRB
284 782A 81 21 CMPA #'1
285 782C 27 E1 BEQ VLIST0
286 782E 54 LSRB
287 782F 81 25 CMPA #'X
288 7831 27 DC BEQ VLIST0
289 7833 54 LSRB
290 7834 81 24 CMPA #'*
291 7836 27 D7 BEQ VLIST0
292 7838 A6 0DFE5D LDA vflag,PCR
293 783C 85 0F BITA #F
294 783E 26 06 BNE VLIST3
295 7840 8A 0F ORA #F

```

## BASIC ANALYZERアセンブル・リスト

296	7842	A7	BDFE53	STA	vflag,PCR	406	7918	10AE	01	LDY	1,X
297	7846	17	01BF	VLIST3	LBSR	V	407	7918	8D	BSR	VAR
298	7849	16	FE79		LBRA	CLOSE	408	7919	8D	BSR	PRIST
299							409	791F	17	FEAC	VS0
300	784C	17	FE08	UN	LBSR	BREAK	410	7922	8D	BSR	PRIKL
301	784F	A6	BDFE46		LDA	vflag,PCR	411	7924	E6	BSR	X
302	7853	95	F0		BITA	#F0	412	7926	10AE	01	LDY
303	7855	27	03		BEQ	VNO	413	7929	8D	BSR	VS0
304	7857	17	FE4D		LBSR	CR	414	792B	8D	BSR	PRIKR
305	785A	A6	84	VNO	LDA	,X	415	792D	30	03	LEAX
306	785C	E6	80		LDB	,X+	416	792F	1F	10	TFR
307	785E	44			LSRA		417	7931	83	0003	SUBD
308	785F	44			LSRA		418				
309	7860	44			LSRA		419	7934	34	02	VADD
310	7861	44			LSRA		420	7936	A6	BDFD5F	PSHS
311	7862	34	02		PSHS	A	421	793A	85	10	LDA
312	7864	34	0F		ANDB	#F0F	422	793C	27	0C	BITA
313	7866	5C			INCB		423	793E	17	FE8D	BEQ
314	7867	A6	80	VNI	LDA	,X+	424	7941	8D	BSR	VARDE
315	7869	17	FE62		LBSR	PRIA	425	7943	A6	E4	BSR
316	786C	5A			DECB		426	7945	17	FD68	LDA
317	786D	25	F8		BNE	VNI	427	7948	8D	BSR	HEXD
318	786F	35	04		PULS	B	428	794A	35	82	BSR
319	7871	86	21		LDA	#'	429				PRIKKR
320	7873	C1	04		CMPB	#4	430	794C	34	04	A,PC
321	7875	27	0C		BEQ	VNE	431	794E	34	10	PSHS
322	7877	4C			INCA		432	7950	EC	81	B
323	7878	4C			INCA		433	7952	34	06	PSHS
324	7879	C1	08		CMPB	#8	434	7954	E6	80	LDB
325	787B	27	06		BEQ	VNE	435	7956	3A		,X+
326	787D	4C			INCA		436	7957	3A		ABX
327	787E	C1	03		CMPB	#3	437	7958	34	14	ABX
328	7880	27	01		BEQ	VNE	438	795A	17	FF78	PSHS
329	7882	4C			INCA		439	795D	34	04	LBSR
330	7883	16	FE48	VNE	LBRA	PRIA	440	795F	EC	83	PRIKL
331							441	7961	83	0001	PSHS
332	7886	17	FF45	V	LBSR	TAB	442	7964	34	10	PSHS
333	7889	34	12		PSHS	A,X	443	7966	17	FD50	LDD
334	788B	108E	043D		LDY	#buftop	444	7969	35	10	SUBD
335	788F	17	0608		LBSR	VCONV	445	796B	6A	E4	#1
336	7892	A6	BDFE03		LDA	vflag,PCR	446	796D	27	05	LBSR
337	7896	85	10		BITA	#F0	447	796F	17	FF69	PULS
338	7898	27	2C		BEQ	VNE	448	7972	20	E8	DEC
339	789A	34	10		PSHS	X	449	7974	32	61	BEQ
340	789C	8E	043D		LDX	#buftop	450	7976	17	FF5F	VAR1
341	789F	17	FE2F		LBSR	WRITE	451	7979	35	14	LEAS



## BASIC ANALYZERアセンブル・リスト

```

516 7A08 9E 35 V LDX simple
517 7A0A 9C 38 V0 CMPX array
518 7A0C 24 39 BCC V1
519 7A0E 8D 28 BSR V03
520 7A10 A5 8DFC85 BITA vflag,PCR
521 7A14 27 17 BEQ V02
522 7A16 6D 01 TST 1,X
523 7A18 2B 13 BMT V02
524 7A1A 17 FE2F LBSR VN
525 7A1D C1 03 CMFB #3
526 7A1F 26 05 BNE V01
527 7A21 17 FEE5 LBSR VS
528 7A24 2D E4 BRA V0
529 7A26 1F 98 V01 TFR B,A
530 7A28 17 FESB LBSR VV
531 7A2B 2D DD BRA V0
532 7A2D A6 84 V02 LDA ,X
533 7A2F 84 0F ANDA #SF
534 7A31 30 85 LEAX A,X
535 7A33 CB 02 ADDB #2
536 7A35 3A ABX
537 7A36 2D 02 BRA V0
538 7A38 A6 84 V03 LDA ,X
539 7A3A 44 LSRA
540 7A3B 44 LSRA
541 7A3C 44 LSRA
542 7A3D 44 LSRA
543 7A3E 1F 89 TFR A,B
544 7A40 81 03 CMPA #3
545 7A42 26 02 BNE V03E
546 7A44 85 01 LDA #1
547 7A46 39 V03E RTS
548 *-----
549 7A47 9E 38 V1 LDX array
550 7A49 9C 3D V11 CMPX free
551 7A4B 24 1E BCC VE
552 7A4D 8D 05 BSR V03
553 7A4F A5 8DFC46 V12 BITA vflag,PCR
554 7A53 26 0E BNE V13
555 7A55 A6 84 LDA ,X
556 7A57 84 0F ANDA #SF
557 7A59 8B 02 ADDA #2
558 7A5B 30 85 LEAX A,X
559 7A5D EC 84 LDD ,X
560 7A5F 30 8B LEAX D,X
561 7A61 20 E5 BRA V11
562 7A63 17 FDE6 V13 LBSR VN
563 7A65 17 FEE3 LBSR VAR
564 7A69 2D 0E BRA V11
565 7A6B 16 FC39 VE LBRA CR
566 *
567 * COLOR TABLE
568 *
569 *
570 7A6E 05 TABLE FCB 5,5,5,1,3,1,0,5
571 7A70 05 FCB 5,5,1,5,15,15,5,*85
572 *
573 7A7E 08 FCB 1,1,1,0,3,3,3,5
574 7A80 06 FCB 6,11,1,5,5,11,11,6
575 *
576 7A8E 04 FCB 4,11,1,4,4,4,4,4
577 7A90 03 FCB 9,9,9,9,2,6,6
578 *
579 7A9E 09 FCB 9,*95,*53,4,12,4,*74,3
580 7AA0 02 FCB 2,*64,11,11,3,3,5,5
581 *
582 7AAE 08 FCB 11,1,*44,4,4,4,1,6
583 7AB0 06 FCB 6,6,9,5,0,5,3
584 *
585 7ABE 00 FCB 0,0,2,0,0,0,*85,0
586 7AC0 05 FCB 5,0,0,0,0,0,0,0
587 *
588 7ACE 00 FCB 0,0,0,0,0,0,0,0
589 7AD0 06 FCB 6,6,6,6,6,6,3,3
590 *
591 7ADE 06 FCB 6,*8F3
592 *
593 *
594 7AE2 00 TABLE FCB 0,0,0,0,1,0,0,0
595 7AEA 00 FCB 0,0,0,0,2,0,0,0
596 *
597 7AF2 00 FCB 0,0,0,0,0,0,0,0
598 7AFA 00 FCB 0,0,0,*21,0,0,0,0
599 *
600 7B02 11 FCB *11,6,2,0,0,1,1,1
601 7B0A 01 FCB 1,*30,0,6,6,6,6,6
602 *
603 7B12 05 FCB *FFB0-FFB4
604 *
605 *
606 *
607 7B17 00 DATA1 FCB 0,*D,0,*A,0,0,5,0
608 7B1F 03 FCB 3,0,0,0,0,*11,*D
609 7B26 14 DATA2 FCB *14,0,0,0,0,0,0,0
610 7B2E 00 FCB 0,0,0,0,*20,0,0,*E
611 7B36 00 FCB 0,0,0,0,*10,*14
612 7B3C 09 DATA3 FCB 9,0,0,*12,3
613 7B41 00 FCB 0,0,0,0,*14,9
614 7B47 07 DATA4 FCB 7,*D,1,0,0,0,0,*14,7
615 *
616 * PRINT CHR$(AccA) with COLOR
617 *
618 7B50 34 76 PRAC PSHS A,B,X,Y,U
619 7B52 6D 8DFB49 TST SN3,PCR
620 7B56 27 05 BEQ PRAC0
621 7B58 17 FB73 LBSR PRIA
622 7B5B 2D 63 BRA PRACE
623 7B5D 30 8CB7 PRAC0 LEAX DATA1,PCR
624 7B60 33 8C11 LEAU <PRAC1,PCR
625 7B63 E6 80 SETB LDB ,X+

```

```

626 7B65 A6 80 SETB0 LDA ,X+
627 7B67 A7 E2 STA ,S
628 7B69 5A DECB
629 7B6A 26 F9 BNE SETB0
630 7B6C E6 84 LDB ,X
631 7B6E 30 66 LEAX 6,S
632 7B70 AF 62 STX 2,S
633 7B72 6E C4 JMP ,U
634 7B74 30 69 PRAC1 LEAX 8,S
635 7B76 AF 62 STX 2,S
636 7B78 30 E4 LEAX ,S
637 7B7A 8D 48 BSR TOBIOS
638 7B7C 10AE 6B LDY #8,S
639 7B7F 32 E5 LEAS 8,S
640 7B81 30 8CA2 LEAX DATA2,PCR
641 7B84 33 8C02 LEAU <PRAC2,PCR
642 7B87 2D DA BRA SETB
643 7B89 1E 20 PRAC2 EXG Y,D
644 7B8B A7 6C STA #C,S
645 7B8D A7 E810 STA #10,S
646 7B8F E7 6E STB #E,S
647 7B92 E7 E812 STB #12,S
648 7B95 1E 02 EXG D,Y
649 7B97 A6 E5 LDA 6,S
650 7B99 A7 6A STA #A,S
651 7B9B 86 01E5 LDA color
652 7B9E A7 69 STA 9,S
653 7BA0 30 E4 LEAX ,S
654 7BA2 8D 23 BSR TOBIOS
655 7BA4 32 E5 LEAS 8,S
656 7BA6 1F 20 TFR Y,D
657 7BA8 4C INCA
658 7BA9 91 C3 CMPA width
659 7BAB 27 15 BEQ PRACCR
660 7BAD 1F 02 TFR D,Y
661 7BAF 30 8C8A LEAX DATA3,PCR
662 7BB2 33 8C02 LEAU <PRAC3,PCR
663 7BB5 2D AC BRA SETB
664 7BB7 10AF 67 PRAC3 STY 7,S
665 7BB9 30 E4 LEAX ,S
666 7BBC 8D 09 BSR TOBIOS
667 7BBE 32 E5 LEAS 8,S
668 7BC0 35 F6 PRACE PULS B,X,Y,U,PC
669 7BC2 17 FAE2 PRACCR LBSR CR
670 7BC5 35 F6 PULS A,B,X,Y,U,PC
671 7BC7 6E 9FFBFA TOBIOS JMP [BIOS]
672 *
673 * READ RANGE
674 *
675 7BCB 6F 8DFAC7 RANGE CLR strno,PCR
676 7BCF 8E 0000 LDX #0
677 7BD2 108E FF9F LDY #FFF9F
678 7BD6 9D D2 JSR getinc
679 7BD8 81 FE CMPA #FE
680 7BDA 27 1D BEQ R2
681 7BDC 81 DA R0 CMA #DA
682 7BDE 27 23 BEQ R3
683 7BE0 81 2C CMA #C
684 7BE2 27 1F BEQ R3
685 7BE4 9F A2 R1 STX start
686 7BE6 109F A4 STY end
687 7BE9 9D D8 JSR get
688 7BED 27 0B BEQ RE
689 7BED 17 FAB4 LBSR STRING
690 7BF0 AF 8DFAA3 STX stradd,PCR
691 7BF4 E7 8DFA9E STB strno,PCR
692 7BF8 39 RE
693 7BF9 8D 1C R2 BSR R5
694 7BF8 1F 01 TFR D,X
695 7BFD 1F 12 TFR X,Y
696 7BFF 9D D2 JSR getinc
697 7C01 2D D9 BRA R0
698 7C03 9D D2 JSR getinc
699 7C05 81 FE CMA #FE
700 7C07 27 0E BEQ R4
701 7C09 108F 9FFF LDY #FFF9F
702 7C0D 2D D5 BRA R1
703 7C0F 8D 06 R4 BSR R5
704 7C11 1F 02 TFR D,Y
705 7C13 9D D2 JSR getinc
706 7C15 2D CD BRA R1
707 7C17 9D D2 R5 JSR getinc
708 7C19 5F CLRBR
709 7C1A 4A DECA
710 7C1B 27 1A BEQ R51
711 7C1D 4A DECA
712 7C1E 27 15 BEQ R52
713 7C20 4A DECA
714 7C21 4A DECA
715 7C22 1026 FBA3 LBNB ERR2
716 7C26 C6 04 R54 LDB #4
717 7C28 9D D2 R55 JSR getinc
718 7C2A 4D TSTA
719 7C2B 1026 FB9A LBNB ERR2
720 7C2F 5A DECB
721 7C30 26 F6 BNE R55
722 7C32 DC A6 LDD current
723 7C34 39 RTS
724 7C35 8D 00 R52 BSR R51
725 7C37 8C DA R51 INC get+2
726 7C39 26 02 BNE R53
727 7C3B 8C D9 INC get+1
728 7C3D A6 9F00D9 R53 LDA [get+1]
729 7C41 1E 89 EXG A,B
730 7C43 39 RTS
731 *
732 * SET COLOR
733 *
734 7C44 86 06 COL6 LDA #6
735 7C46 8C FCB CMPX

```

[illegible]

## BASIC ANALYZERアセンブル・リスト

```

956 7E12 20 E7 BRA SETC3
957 7E14 C1 28 SPEN CMPB #<
958 7E16 27 E3 BEQ SETC3
959 7E18 20 DC BRA SETC1
960 7E1A C1 22 STIME CMPB #<
961 7E1C 26 E0 BNE SETCE
962 7E1E F6 01E5 LDB color
963 7E21 C1 04 CMPB #4
964 7E23 27 D3 BEQ SETCE
965 7E25 20 CF BRA SETC1
966 7E27 C1 40 SPUT CMPB #0
967 7E29 27 D0 BEQ SETC3
968 7E2B 20 CC BRA SETC2
969 7E2D C1 28 SKEY CMPB #<
970 7E2F 26 CA BNE SETC3
971 7E31 20 C3 BRA SETC1
972 7E33 C1 23 SPRI CMPB #<
973 7E35 26 CA BNE SETC3
974 7E37 20 C0 BRA SETC2
975 7E39 C1 FF SLIN CMPB #FFF
976 7E3B 26 BE BNE SETC3
977 7E3D 06 01 LDA #1
978 7E3F 20 BA BRA SETC3
979 7E41 C1 FE STHEN CMPB #FFE
980 7E43 27 BE BEQ SETC3
981 7E45 A7 8DF851 STA flag2,PCR
982 7E49 20 B0 BRA SETC3
983 7E4B E6 84 SCOM LDB #X
984 7E4D C1 A2 CMPB #A2 MON
985 7E4F 26 AA BNE SETC3
986 7E51 06 00 LDA #0
987 7E53 20 A6 BRA SETC3
988
989 7E55 00 CJUMP FCB SSCR-SSCR,SPEN-SSCR
990 7E57 0E FCB STIME-SSCR,SPUT-SSCR
991 7E59 21 FCB SKEY-SSCR,SPRI-SSCR
992 7E5B 20 FCB SLIN-SSCR,STHEN-SSCR
993 7E5D 3F FCB SCOM-SSCR
994
995 7E5E CE 01E5 CLIST5 LDU #table-10
996 7E61 81 FF CMPA #FFF FUNCTION
997 7E63 26 04 BNE CLIS51
998 7E65 A6 1F LDA -1,X
999 7E67 33 45 LEAU 5,U
1000 7E69 84 7F CLIS51 ANDA #07F
1001 7E6B 33 4A CLIS52 LEAU 10,U
1002 7E6D 60 C4 TST ,U
1003 7E6F 1027 FF23 LBEO CLIS34
1004 7E73 A0 C4 SUBA ,U
1005 7E75 2A F4 BPL CLIS52
1006 7E77 A0 C4 ADDA ,U
1007 7E79 EE 41 LDU 1,U TABLE TOP
1008 7E7B A4 C4 CLIS53 DECA
1009 7E7D 28 06 BMI CLIS55
1010 7E7F 60 C0 CLIS54 TST ,U+
1011 7E80 2A FC BPL CLIS54
1012 7E82 20 F7 BRA CLIS53
1013 7E84 A6 C4 CLIS55 LDA ,U
1014 7E86 84 7F ANDA #07F
1015 7E88 17 0098 LBSR CLOUT
1016 7E8B 60 C0 TST ,U+
1017 7E8D 2A F5 BPL CLIS55
1018 7E8F 6D 8DF807 TST flag2,PCR
1019 7E93 1027 FEBC LBEO CLIS21
1020 7E97 16 FEF0 LBRA CLIS32
1021
1022 7E9A 34 82 VCONV PSHS A
1023 7E9C 81 01 CMPA #01 1BYTE
1024 7E9E 27 1A BEQ VCON11
1025 7EA0 81 02 CMPA #02 2BYTE
1026 7EA2 27 1A BEQ VCON12
1027 7EA4 81 F2 CMPA #F2 ?
1028 7EA6 26 09 BNE VCONV1 4or8 BYTE
1029 7EA8 EC 81 LDD ,X++
1030 7EAA 34 30 PSHS X,Y
1031 7EAC 17 F810 LBSR F2SET1
1032 7EAF 20 1B BRA VCONV2
1033 7EB1 97 17 VCONV1 STA cstype
1034 7EB3 17 F800 LBSR DV4SET
1035 7EB5 30 86 LEAX A,X
1036 7EB7 20 10 BRA VCON14
1037 7EBA 0F 76 VCON11 CLR cset
1038 7EBC 20 04 CLR VCON13
1039 7EBE A6 80 VCON12 LDA ,X+
1040 7EC0 97 76 STA cset
1041 7EC2 86 02 VCON13 LDA #02 (2byte)
1042 7EC4 97 17 STA cstype
1043 7EC6 A6 80 LDA ,X+
1044 7EC8 97 77 STA cset+1
1045 7ECA 34 30 VCON14 PSHS X,Y
1046 7ECC 17 F7ED VCONV2 LBSR CONV
1047 7ECF 35 40 PULS U
1048 7ED1 35 20 PULS Y
1049 7ED3 35 04 PULS B
1050 7ED5 34 40 PSHS U
1051 7ED7 6F 8DF7BF CLR flag2,PCR
1052 7EDB 6F 8DF7BC CLR flag3,PCR
1053 7EDF A6 80 VCON21 LDA ,X+
1054 7EE1 91 20 CMPA #<
1055 7EE3 27 FA BEQ VCON21
1056 7EE5 80 3C VCON22 BSR CLOUT
1057 7EE7 81 2E CMPA #<
1058 7EE9 26 86 BNE VCON23
1059 7EEB 6C 8DF7AB INC flag2,PCR
1060 7EEF 20 8C BRA VCON25
1061 7EF1 81 45 VCON23 CMPA #<
1062 7EF3 27 04 BEQ VCON24
1063 7EF5 81 44 CMPA #<
1064 7EF7 26 84 BNE VCON25
1065 7EF9 6C 8DF79E VCON24 INC flag3,PCR

```

```

1066 7EFD A6 80 VCON25 LDA ,X+
1067 7EFF 26 E4 BNE VCON22
1068 7F01 6D 8DF79E TST flag3,PCR
1069 7F05 26 1A BNE VCON28
1070 7F07 6D 8DF79F TST flag2,PCR
1071 7F0B 26 0C BNE VCON26
1072 7F0D C1 F2 CMPB #0F2 ?
1073 7F0F 27 10 BEQ VCON28
1074 7F11 C1 04 CMPB #04 (4byte)
1075 7F13 26 04 BNE VCON26
1076 7F15 86 21 LDA #<
1077 7F17 20 06 BRA VCON27
1078 7F19 C1 08 VCON26 CMPB #08 (8byte)
1079 7F1B 26 04 BNE VCON28
1080 7F1D 86 23 LDA #<
1081 7F1F 80 02 VCON27 BSR CLOUT
1082 7F21 35 90 VCON28 PULS X,PC
1083
1084 7F23 108C 053C CLOUT CMFY #bufend
1085 7F27 24 10 BCC LCOUTE
1086 7F29 A7 A0 STA ,Y+
1087 7F2B 6F A4 CLR ,Y
1088 7F2D 6D 8DF76D TST SW2,PCR
1089 7F31 26 0A BNE SW2A
1090 7F33 6D 8DF766 TST SW1,PCR
1091 7F37 27 01 BEQ SW1A
1092 7F39 39 LCOUTE RTS
1093
1094 7F3A 17 FC13 SW1A LBSR PRAC
1095 7F3D 6D 8DF755 SW2A TST strno,PCR
1096 7F41 27 F6 BEQ LCOUTE
1097 7F43 34 36 PSHS A,B,X,Y
1098 7F45 E6 8DF74D LDB strno,PCR
1099 7F49 AE 8DF74A LDX stradd,PCR
1100 7F4D 3A ABX
1101 7F4E A6 82 LDA ,X+
1102 7F50 A1 A2 CMPA ,Y-
1103 7F52 26 09 BNE COMPE
1104 7F54 5A DECB
1105 7F55 27 08 BEQ BEEP
1106 7F57 108C 043D CMFY #bufstop
1107 7F5B 22 F1 BHI COMF
1108 7F5D 35 B6 COMPE PULS A,B,X,Y,PC
1109
1110 7F5F 8E 1000 BEEP LDX #1000
1111 7F62 86 81 LDA #01
1112 7F64 A7 8DF734 STA flag4,PCR
1113 7F68 6D 8DF732 TST SW2,PCR
1114 7F6C 26 EF BNE COMPE
1115 7F6E 87 FD03 STA BEL
1116 7F71 3D BEEP1 MUL
1117 7F72 30 1F LEAX -1,X
1118 7F74 26 FB BNE BEEP1
1119 7F76 87 FD03 STA BEL
1120 7F79 20 E2 BRA COMPE
1121
1122 * TITLE
1123
1124 7F7B 34 40 NAME PSHS U
1125 7F7D 17 F754 LBSR CLS
1126 7F80 30 8C04 LEAX <NAME1-1,PCR
1127 7F83 17 F724 LBSR WRITE2
1128 7F86 35 C0 PULS U,PC
1129
1130 7F88 2A NAME1 FCC /***** BASIC ANALYZE/
1131 7F9C 52 FCC /R Version 2.0 *****/
1132 7F80 0D FCC 13,10
1133 7F82 20 FCC / Expanded => CLIST/
1134 7FC4 2C FCC /,VLIST,FIND,SYSTEM/
1135 7FD6 0A FCC 10,00
1136
1137 * FM-7 JUMP TABLE
1138
1139 7FD8 80D1 JTABLE FDB $80D1,$98F1,$9850,$980B
1140 7FE0 9C22 FDB $9C22,$AC37,$AC3D,$B301
1141 7FE8 B615 FDB $B615,$B620,$974D,$8D19
1142 7FF0 CE4B FDB $CE4B,$CEDF,$D072,$D08E
1143 7FF8 D90F FDB $D90F,$DA80,$D059,$D678
1144
1145 8000 0000 END RMB 0
1146
1147
1148 * CHECK SUM PROGRAM
1149 * P=0:FOR I=&H7600 TO &H7FFF:P=
1150 * P+PEEK(I):NEXT I:IF P[RETURN]
1151 * CHECK SUM => 247621
1152
1153 END

```

# F-BASICに プロット・コマンドを!

■ ROBOCON

『F-BASIC EXPANDER』を使ってみて、その便利さに感激してしまいました。

そこで、私もプロット関係の拡張命令を作ったので、発表します。

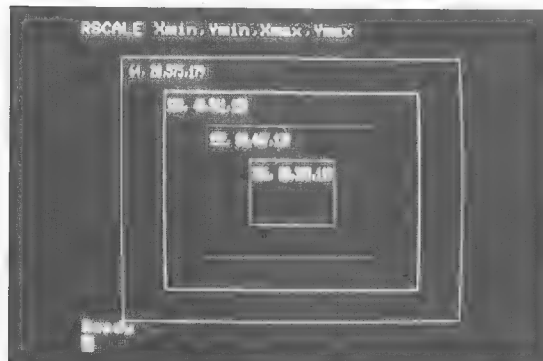
## 目 的

いままでもFM-8用として、グラフ作成関係のプログラムが発表されていますが、機能はなかなか便利なのに、BASICで書かれているためいろいろと不便な面がありました。そこで、それらの一部ですが、拡張命令として実現してみました。

まず、拡張される命令の考え方としては、図1のように、任意の仮想画面(VXmin, VYmin)~(VXmax, VYmax)を定義し、その座標をFM-8の実画面である、640×200ドットの内(RXmin, RYmin)~(RXmax, RYmax)に変換してやろうというものです。つまり、FM-8の画面に任意の座標が定義でき、その座標でプロットなどができるという訳です。これらについては、参考文献2)にわかりやすく書かれているので、もっと詳しく知りたい方はそちらの方を参考にしてください。

## 仕 様

拡張される命令は座標の定義、プロットのステートメントが6個、座標の読み取り、フォーマット付文字変換の関  
写真1 テスト・プログラム実行例



数が3個です。

以下、これらの使い方などを説明します。

**VSCALE Xmin, Ymin, Xmax, Ymax**

引数：F-BASICの整数、単精度数式の範囲。倍精度数式でもかまいませんが、内部で単精度に変換されてしまいます。

説明：仮想画面の座標を定義する命令で、Xminなどは図1のVXminなどに対応します。なお、リスト1のプログラムをロードした直後は、VSCALE 0, 0, 639, 199に設定されています。

**RSCALE Xmin, Ymin, Xmax, Ymax**

引数：正の整数の範囲。しかし、FM-8の画面が640×200ドットですから、X側が639を、Y側が199を超えると意味を持ちません。

説明：実画面の使用範囲などを定義する命令で、Xminなどは図1のRXminなどに対応します。なお、この命令はFM-8が640×200ドットの分解能であるため、あまり機能が発揮できないようです。

リスト1のプログラムをロードした直後は、RSCALE 0, 199, 639, 0に設定されています。

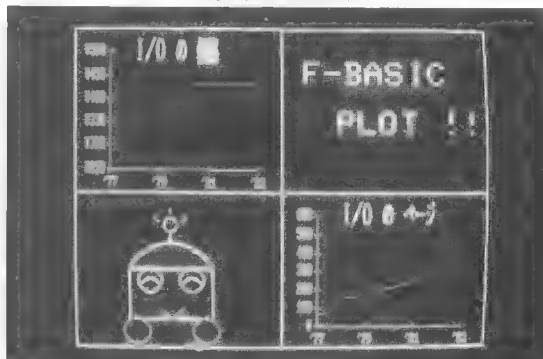
つまり、画面の左下が原点(0, 0)になります。

**MOVA X, Y**

引数：F-BASICの整数、単精度数式で、VSCALEで定義した範囲内。

説明：現在の位置(以前に移動、または描画した後の位置)

写真2 応用例



から (X, Y) の位置まで線を描かないで移動します。

**DRWA X, Y[[, 機能][, カラー]]**

引数: XとYはMOVAと同じで、機能はPSET, PRESET, AND, OR, XORのいずれかです。カラーは0～7のコードです。

説明: 現在の位置から (X, Y) の位置まで、指定した機能および、カラーで線を描きます。なお、機能または、カラーを省略した場合は以前に指定した機能、カラーが使われます。

**MOVR X, Y**

引数: F-BASICの整数, 単精度数式で、加えた結果が、VSCALEで定義した範囲内。

説明: 現在の位置に (X, Y) を加えた位置まで移動。

**DRWR X, Y[[, 機能][, カラー]]**

引数: XとYはMOVRと同じで、機能およびカラーはDRWAと同じです。

説明: 現在の位置から、現在の位置 (X, Y) を加えた位置まで、指定した機能、カラーで線を描きます。

**RPOSX([X])**

引数: F-BASICの整数, 単精度数式で、VSCALEで定義した範囲内。

説明: 仮想座標のXを与えると、そのときの実座標での値を返してきます。Xを省略すると現在の位置での実座標の値を返してきます。

**RPOSY([Y])**

説明: XがYに変わるだけでRPOSXと同じです。

**FSTR\$(フォーマット文字列, 数式)**

引数: フォーマット文字列は“#”, “.”, “+”, “-”, “<”などを使って、数値のフォーマットを指定します。なお, “<”は1個のみでよく、F-BASICのように4個を並べる必要はありません。数式はF-BASICの整数, 単精度, 倍精度数式が使えます。

説明: 数式の値をフォーマット文字列の書式指定に従い、文字列に変換する関数で、F-BASICのPRINT USINGのようなものです。ただ、数値のフォーマットのみで、文字、空白などのフォーマットはできません。

また、フォーマット文字列に、“\*”, “¥”, “,”なども一応使えますが、“\*”を除き、それ自身の桁領域が確保されないで、“#”で桁領域を確保する必要があります。

実行例を図2に示しますので、参考にしてください。

図1 仮想画面と実画面

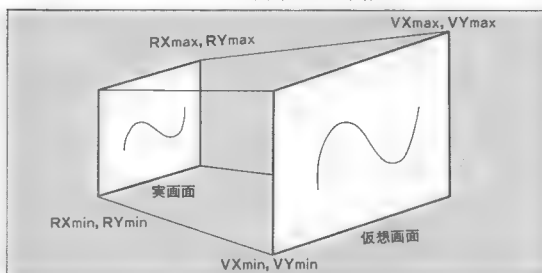


図2 FSTR\$の実行例

入力数値	フォーマット文字	出力文字
12.3456	#####.##	12.35
1234	+#####.##	+1234.00
-12345	-#####	12345-
123456	#####<	12.346E+04
12345	¥, #####	¥¥¥12,345

## プログラムの起動

リスト1のプログラムをアセンブラまたはモニタを使って入力してください。このプログラムはディスク・システムで使うことを想定しているので、ROMベースで使う場合は次の変更をしてください。

5A19: 02→01                      5A1A: 03→F9  
5A41: EE→E8                      5A63: 67→56

なお、このプログラムはリロケータブルなので適当な番地において、その先頭番地をEXECで実行してください。

## 終わりに

リスト2に簡単なテスト・プログラムを示します。

写真1がその実行結果で、RSCALEで定義したときの範囲を表わしています。また、写真2は応用例でRSCALEによって画面の1/4を設定し、グラフ、文字、絵などを描いてみました。グラフなどはおもに、DRWAを、文字、絵などはRPOS関数で位置を求めて描きました。

### 参考文献

- 1) 笠作貴弥: “F-BASIC EXPANDER” I/O'82年9月号
- 2) 伊藤 誠他: “図形を描くためのソフトウェア技術” インターフェイス'79年11月号
- 3) 世田谷マイコンクラブ: “FM-8 システム・エリアの解析” I/O'81年12月号, I/O別冊FM-8活用研究

リスト1 F-BASIC PLOTアセンブル・リスト

```

0000
0001
0002
0003
0004
0005 5A00
0006
0007 5A00 30 8C FB
0008 5A03 9C 45
0009 5A05 24 11
0010 5A07 DC 45
0011 5A09 93 3F
0012 5A0B 34 06
0013 5A0D 9F 45
0014 5A0F 1F 10
0015 5A11 A3 E1
0016 5A13 DD 3F
0017 5A15 BD 9414

*****
* F-BASIC Expander SFL0T Ver2.2 *
* 1982.12.15 *
*****
*
* ORG $5A00
* BEGIN LEAX <BEGIN-2,PCR
          CMPL <#45
          BCC SETCOD
          LDD <#45
          SUBD <#3F
          PSHS E,A
          STX <#45
          TFR X,D
          SUBD S++
          STD <#3F
          JSR $9414

character top
keyword set
string bottom
character top
:
string bottom
all value clear

```

## リスト1 F-BASIC PLOT アセンブル・リスト

```

0018 5A18 CE 0203 SETCOD LDU #203 set keyword
0019 5A1B 86 06 LDA #6 "イカ メイレー スク
0020 5A1D A7 C0 STA U+
0021 5A1F 30 8C 5D LEAX <CODE,PCR : set CODE
0022 5A22 AF C1 STX ,U++
0023 5A24 30 8C 19 LEAX <ANALYS,PCR : set ANALYS
0024 5A27 AF C1 STX ,U++
0025 5A29 86 03 LDA #3 "イカ カンスウ
0026 5A2B A7 C0 STA U+
0027 5A2D 30 8C 6C LEAX <FCODE,PCR : set FCODE
0028 5A30 AF C1 STX ,U++
0029 5A32 30 8C 2D LEAX <FANLYS,PCR : set FANLYS
0030 5A35 AF C4 STX ,U
0031 5A37 30 8C 71 LEAX <MSG-1,PCR set MSG
0032 5A3A BD 9FDF JSR $9FDF line out
0033 5A3D 7E 948D JMP $948D BASIC hot
0034
0035 5A40 80 EE * ANALYS SUBA #EE EX-code min
0036 5A42 1027 0080 LBEO VSCALE
0037 5A46 4A DECA
0038 5A47 1027 00AD LBEO RSCALE
0039 5A4B 4A DECA
0040 5A4C 1027 0103 LBEO MOVA
0041 5A50 4A DECA
0042 5A51 1027 0102 LBEO DRWA
0043 5A55 4A DECA
0044 5A56 1027 0117 LBEO MOVR
0045 5A5A 4A DECA
0046 5A5B 1027 0116 LBEO DRWR
0047 5A5F 7E 9740 SERR3 JMP $9740 syntax error
0048
0049 5A62 C0 67 * FANLYS SUBB #67 code min
0050 5A64 25 F9 BCS SERR3 syntax error
0051 5A66 57 CMPE #3
0052 5A67 C1 03 BCC SERR3 code スウ-1
0053 5A69 24 F4 BCS SERR3 syntax error
0054 5A6B 34 04 PSHS
0055 5A6D BD 972F JSR $972F ' ' test
0056 5A70 35 04 PULS B
0057 5A72 5D TSTB
0058 5A73 1027 01D3 LBEO RPOX
0059 5A77 5A DECB
0060 5A7B 1027 01D1 LBEO RPOSY
0061 5A7C 16 0210 LBRA FSTR FSTR$
0062
0063 5A7F 56 * CODE FCC /VSCALE+ VSCALE
0064 5A85 52 FCC /RSCALE+ RSCALE
0065 5A8B 4D FCC /MOV# MOV
0066 5A8F 4A FCC /DRW# DRWA
0067 5A93 4D FCC /MOV# MOVR
0068 5A97 44 FCC /DRW# DRWR
0069 5A9B 00 FCB $0
0070
0071 5A9C 52 * FCODE FCC /RPOX+ RPOX()
0072 5AA1 52 FCC /RPOSY+ RPOSY()
0073 5AA6 46 FCC /FSTR+ FSTR$()
0074 5AAB 00 FCB $0
0075
0076 5AAC 46 * MSG FCC /F-BASIC/
0077 5AB3 20 FCC / Expanded/
0078 5ABC 20 FCC / PLOT !!!/
0079 5AC5 00 FCB $0
0080
0081 5AC6 9D D2 * VSCALE JSR <$D2 get next
0082 5ACB C6 04 LDB #4
0083 5ACA 34 04 PSHS B
0084 5ACC 30 8D 029B LEAX <VXMIN,PCR set counter
0085 5AD0 8D 67 BSR FRMTCM set VXMIN
0086 5AD2 8D 7C BSR FACUL1 formula ,check
0087 5AD4 30 04 LEAX 4,X save input data
0088 5AD6 6A E4 DEC S pointer to next
0089 5ADB 26 F6 BNE INPUTV count down
0090 5ADA 8D 6F BSR CHKEND
0091 5ADC C6 02 LDB #2 end check
0092 5ADE E7 E4 STB S
0093 5AD0 30 14 LEAX -12,X
0094 5AE2 31 08 LEAY 8,X set counter
0095 5AE4 17 00FE LBSR LDFAC set VYMIN
0096 5AE7 17 010F LBSR CSUB set V*MAX
0097 5AEA 8D 64 BSR FACUL1 load V*MAX
0098 5AEC 6D 84 BSR to V*L
0099 5AEE 27 43 BEQ IERR1 ? 0
0100 5AF0 30 14 LEAX -2,X
0101 5AF2 6A E4 DEC S set VXMIN
0102 5AF4 26 EC BNE VRANGE count down
0103 5AF6 20 35 BRA SCALE1
0104
0105 5AF8 9D D2 * RSCALE JSR <$D2 get next
0106 5AFA 33 8D 0289 LEAU RXMIN,PCR set RXMIN
0107 5AFE C6 04 LDB #4
0108 5B00 34 04 PSHS B
0109 5B02 34 40 LEAX U set counter
0110 5B04 BD 9E8F JSR $9E8F formula to INT
0111 5B07 E6 62 LDB 2,S
0112 5B09 5A DECB
0113 5B0A 27 03 BEQ PAS1 check counter
0114 5B0C BD 9732 JSR $9732 comma test
0115 5B0F 35 40 PULS U
0116 5B11 9E 76 LDX <$76
0117 5B13 AF C1 STX ,U++
0118 5B15 6A E4 DEC S load FAC1
0119 5B17 26 E9 BNE INPUTR save input data
0120 5B19 8D 30 BSR CHKEND count down
0121 5B1B C6 02 LDB #2 end check
0122 5B1D E7 E4 STB S
0123 5B1F EC C3 LDD ,--U set counter
0124 5B21 A3 5C SUBD -4,U load R*MAX
0125 5B23 29 11 BVS OERR1 R*MAX - R*MIN
0126 5B25 ED 5B STD -8,U over flow
0127 5B27 27 0A BEQ IERR1 to R*L
? 0

```

リスト1 F-BASIC PLOT アセンブル・リスト

0128	5B29	6A	E4	DEC	S	counter down
0129	5B2B	26	F2	BNE	RRANGE	
0130	5B2D	16	00BD	SCALE1	LBRA	SCALE
0131				*		CALQL scale
0132	5B30	7E	9740	SERR1	JMP	\$9740
0133				*		syntax error
0134	5B33	7E	9B03	IERR1	JMP	\$9B03
0135				*		illigal
0136	5B36	7E	B3B8	OERR1	JMP	\$B3B8
0137				*		over flow
0138	5B39	34	10	FRMTCM	PSHS	X
0139	5B3B	BD	975F	JSR	\$975F	formura, test
0140	5B3E	BD	C04A	JSR	\$C04A	to single
0141	5B41	E6	64	LDB	4, S	load counter
0142	5B43	5A		DECB		? 1
0143	5B44	27	03	BEQ	PAS2	
0144	5B46	BD	9732	JSR	\$9732	comma test
0145	5B49	35	90	PAS2	PULS	X, PC
0146				*		
0147	5B4B	9D	D8	CHKEND	JSR	<\$D8
0148	5B4D	26	E1	BNE	SERR1	
0149	5B4F	39		RTS		get now
0150				*		syntax error
0151	5B50	7E	B6AC	FACUL1	JMP	\$B6AC
0152				*		FAC unload
0153	5B53	0F	C4	MOVA	CLR	<PEN
0154	5B55	20	04	BRA	INPA	
0155				*		pen off
0156	5B57	86	01	DRWA	LDA	#1
0157	5B59	97	C4	STA	<PEN	pen on
0158	5B5B	9D	D2	INPA	JSR	<\$D2
0159	5B5D	C6	02	LDB	#2	get next
0160	5B5F	34	04	PSHS	B	set counter
0161	5B61	30	BD	01FE	VPX, PCR	set VPX
0162	5B65	BD	D2	BSR	FRMTCM	formura, test
0163	5B67	BD	E7	BSR	FACUL1	save input data
0164	5B69	30	04	LEAX	4, X	pointer to next
0165	5B6B	6A	E4	DEC	S	caunter down
0166	5B6D	26	F6	BNE	INPA1	
0167	5B6F	20	20	BRA	EVPOPT	
0168				*		evaluat of OPT
0169	5B71	0F	C4	MOVR	CLR	<PEN
0170	5B73	20	04	BRA	INPR	
0171				*		pen off
0172	5B75	86	01	DRWR	LDA	#1
0173	5B77	97	C4	STA	<PEN	pen on
0174	5B79	9D	D2	INPR	JSR	<\$D2
0175	5B7B	C6	02	LDB	#2	get next
0176	5B7D	34	04	PSHS	B	set counter
0177	5B7F	30	BD	01EO	VPX, PCR	set VPX
0178	5B83	BD	B4	BSR	FRMTCM	formura, test
0179	5B85	BD	64	BSR	LDFAC2	load VP*
0180	5B87	BD	77	BSR	CADD	VP* + R*
0181	5B89	BD	C5	BSR	FACUL1	save input data
0182	5B8B	30	04	LEAX	4, X	pointer to next
0183	5B8D	6A	E4	DEC	S	caunter down
0184	5B8F	26	F2	BNE	INPR1	
0185	5B91	32	61	LEAS	1, S	
0186	5B93	9D	D8	JSR	<\$D8	get now
0187	5B95	27	1E	BEQ	TOREAL	to real
0188	5B97	BD	9732	JSR	\$9732	comma test
0189	5B9A	C6	05	LDB	#5	set num-type
0190	5B9C	BD	E0D3	JSR	\$E0D3	line type
0191	5B9F	25	09	BCS	OPT2	evaluat OPT2
0192	5BA1	D7	C2	STB	<\$C2	set line type
0193	5BA3	9D	D2	JSR	<\$D2	get next
0194	5BA5	27	0E	BEQ	TOREAL	to real
0195	5BA7	BD	9732	JSR	\$9732	comma test
0196	5BA9	BD	9E49	JSR	\$9E49	formura to B
0197	5BAC	BD	9C	BSR	CHKEND	end check
0198	5BAF	C1	08	CMFB	#8	
0199	5BB1	24	07	BCC	IERR	illigal
0200	5BB3	D7	FB	STB	<\$FB	set color
0201	5BB5	BD	5E	TOREAL	BSR	CALQLT
0202	5BB7	16	0162	LBRA	PLOT	to calculate
0203				*		to PLOT
0204	5BBA	7E	9B03	IERR	JMP	\$9B03
0205				*		illigal
0206	5BBD	32	61	SCALE	LEAS	1, S
0207	5BBF	4F		CLRA		delete counter
0208	5BC0	5F		CLRB		
0209	5BC1	DD	C9	STD	<\$C9	D <= 0
0210	5BC3	DD	CB	STD	<\$CB	clear end X
0211	5BC5	30	BD	01BA	RXL, PCR	clear end Y
0212	5BC9	31	10	LEAX	-16, X	set RXL
0213	5BCB	BD	04	BSR	SCALES	set VXL
0214	5BCD	30	0A	LEAX	10, X	VXL/RXL=>BAIX
0215	5BCF	31	12	LEAX	-14, X	set RYL
0216	5BD1	C6	02	LDB	#2	set VYL
0217	5BD3	BD	1F	BSR	LDFAC1	INT value set
0218	5BD5	34	20	PSHS	Y	load RXL
0219	5BD7	BD	C04A	JSR	\$C04A	
0220	5BDA	35	10	PULS	X	to single
0221	5BDC	BD	0D	BSR	LDFAC2	
0222	5BDE	BD	27	BSR	CDIV	load VXL
0223	5BE0	30	08	LEAX	8, X	VXL / RXL
0224	5BE2	7E	B6AC	FACUL2	JMP	\$B6AC
0225				*		set BAIX
0226	5BE5	C6	04	LDFAC	LDB	#4
0227	5BE7	BD	0B	BSR	LDFAC1	set single
0228	5BE9	1F	21	TFR	Y, X	load FAC1
0229	5BEB	C6	04	LDFAC2	LDB	#4
0230	5BED	D7	17	STB	<\$17	set single
0231	5BEF	0F	15	CLR	<\$15	
0232	5BF1	7E	B52E	JMP	\$B52E	clear # flag
0233				*		load FAC2
0234	5BF4	D7	17	LDFAC1	STB	<\$17
0235	5BF6	7E	B676	LDFAC1S	JMP	\$B676
0236				*		set type
0237	5BF9	34	10	CSUB	PSHS	load FAC1
					X	



## リスト1 F-BASIC PLOT アセンブル・リスト

0238	5BFB	BD	C0A9		JSR	\$C0A9		FAC2 - FAC1
0239	5BFE	35	90		PULS	X,PC		
0240								
0241	5C00	34	10	* CADD	PSHS	X		
0242	5C02	BD	C0B7		JSR	\$C0B7		FAC2 + FAC1
0243	5C05	35	90		PULS	X,PC		
0244								
0245	5C07	34	10	* CDIV	PSHS	X		
0246	5C09	BD	9871		JSR	\$9871		FAC2 / FAC1
0247	5C0C	35	90		PULS	X,PC		
0248								
0249	5C0E	34	10	* CINT	PSHS	X		
0250	5C10	BD	BFDD		JSR	\$BFDD		FAC1 to INT
0251	5C13	35	90		PULS	X,PC		
0252								
0253	5C15	C6	02	* CALQLT	LDB	#2		
0254	5C17	34	04		PSHS	B		
0255	5C19	30	8D	O14E	LEAX	VXMIN,PCR		set counter
0256	5C1D	31	18	CALQ1	LEAX	-8,X		set VXMIN
0257	5C1F	8D	C4		BSR	LDFA1		set VP*
0258	5C21	8D	D6		BSR	CSUB		
0259	5C23	30	88	18	LEAX	24,X		VP* - VXMIN
0260	5C26	34	10		PSHS	X		set BAI*
0261	5C28	BD	B6F5		JSR	\$B6F5		
0262	5C2B	35	10		PULS	X		FAC1 => FAC2
0263	5C2D	8D	C7		BSR	LDFA1S		
0264	5C2F	8D	D6		BSR	CDIV		load BAI*
0265	5C31	8D	DB		BSR	CINT		/ BAI*
0266	5C33	E6	E4		LDB	,S		to integer
0267	5C35	5A			DECB			load counter
0268	5C36	26	02		BNE	PAS3		
0269	5C38	30	1E		LEAX	-2,X		? 1
0270	5C3A	ED	76	PAS3	LDD	<\$76		
0271	5C3C	E3	06		ADD	12,X		load FAC1
0272	5C3E	ED	88	14	STD	20,X		R*MIN
0273	5C41	30	14		LEAX	-12,X		to ABS*
0274	5C43	6A	E4		DEC	S		set VYMIN
0275	5C45	26	D6		BNE	CALQ1		counter down
0276	5C47	32	61		LEAS	1,S		
0277	5C49	39			RTS			
0278								
0279	5C4A	5F		* RPOSX	CLRB			
0280	5C4B	20	02		BRA	RPOS		
0281								
0282	5C4D	C6	02	* RPOSY	LDB	#2		
0283	5C4F	30	8D	O110	LEAX	VPX,PCR		set VPX
0284	5C53	33	1C	RPOS	LEAU	-4,X		set SFAC
0285	5C55	3A			ABX			
0286	5C56	1F	12		TFR	X,Y		ABS* point
0287	5C58	3A			ABX			set VP*
0288	5C59	EC	84		LDD	,X		load VP* high
0289	5C5B	ED	C4		STD	,U		to SFAC
0290	5C5D	EC	02		LDD	2,X		load VP* low
0291	5C5F	ED	42		STD	2,U		to SFAC
0292	5C61	34	70		PSHS	U,Y,X		
0293	5C63	9D	D8		JSR	<\$D8		get now
0294	5C65	81	29		CMPA	##29		')
0295	5C67	27	0D		BEQ	RPOS1		ヒキスウ ナシ
0296	5C69	BD	975F		JSR	\$975F		formura
0297	5C6C	BD	C04A		JSR	\$C04A		CSNG
0298	5C6F	35	10		PULS	X		
0299	5C71	BD	B6AC		JSR	\$B6AC		to VP*
0300	5C74	34	10		PSHS	X		
0301	5C76	BD	972C	RPOS1	JSR	\$972C		')
0302	5C79	BD	9A		BSR	CALQLT		'test
0303	5C7B	35	70		PULS	X,Y,U		cov. ABS
0304	5C7D	EC	AB	2C	LDD	44,X		
0305	5C80	DD	76		STD	<\$76		load ABS*
0306	5C82	86	02		LDA	#2		to FAC
0307	5C84	97	17		STA	<\$17		set INT
0308	5C86	EC	C4		LDD	,U		
0309	5C88	ED	84		STD	,X		load SFAC
0310	5C8A	EC	42		LDD	2,U		to VP*high
0311	5C8C	ED	02		STD	2,X		load SFAC
0312	5C8E	39			RTS			to VP*low
0313								
0314	5C8F	81	2C	* FSTR	CMPA	##2C		? , ,
0315	5C91	27	4C		BEQ	NEXT		ショウリナク
0316	5C93	BD	9DB6		JSR	\$9DB6		formura STR
0317	5C96	34	14		PSHS	X,B		
0318	5C98	4F			CLRA			
0319	5C99	5F			CLRB			
0320	5C9A	DD	B8		STD	<\$B8		clear USING
0321	5C9C	97	BA		STA	<\$BA		
0322	5C9E	35	14		PULS	B,X		
0323	5CA0	5D			TSTB			check CHR
0324	5CA1	27	3C		BEQ	NEXT		
0325	5CA3	34	04	FSTR	PSHS	B		
0326	5CA5	D6	BA		LDB	<\$BA		load flag
0327	5CA7	A6	80		LDA	,X+		:
0328	5CA9	81	23		CMPA	##23		"#"
0329	5CAB	27	1F		BEQ	SHARP		
0330	5CAD	81	2C		CMPA	##2C		" , "
0331	5CAF	27	30		BEQ	COMMA		" * "
0332	5CB1	81	2A		CMPA	##2A		" * "
0333	5CB3	27	30		BEQ	ASTA		" * "
0334	5CB5	81	5C		CMPA	##5C		" * "
0335	5CB7	27	30		BEQ	YEN		" + "
0336	5CB9	81	2B		CMPA	##2B		" - "
0337	5CBB	27	30		BEQ	PLUS		" - "
0338	5CBD	81	2D		CMPA	##2D		" - "
0339	5CBF	27	30		BEQ	MINUS		" - "
0340	5CC1	81	5E		CMPA	##5E		" - "
0341	5CC3	27	30		BEQ	EXP		" - "
0342	5CC5	81	2E		CMPA	##2E		" - "
0343	5CC7	27	30		BEQ	TEN		
0344	5CC9	7E	9B03		JMP	\$9B03		illigal
0345								
0346	5CCC	0D	B8	* SHARP	TST	<\$B8		USING low
0347	5CCE	26	04		BNE	SHARP1		

## リスト1 F-BASIC PLOT アセンブル・リスト

```

0348 5CD0 0C B9      INC    <$B9
0349 5CD2 20 02      BRA    SHARP2      USING high
0350                  *
0351 5CD4 0C B8      SHARP1 INC    <$B8      USING low
0352 5CD6 CA B0      SHARP2 ORB    #$B0      fmt set
0353 5CD8 D7 BA      FSTR1  STB    <$BA      set flag
0354 5CDA 35 04      PULS   B
0355 5CDC 5A        DECB   B
0356 5CDD 26 C4      BNE    FSTR1
0357 5CDF 20 20      NEXT   BRA    NEXT1
0358                  *
0359 5CE1 CA C0      *COMMA ORB    #$C0      ', ' set
0360 5CE3 20 F3      BRA    FSTR1
0361                  *
0362 5CE5 CA A0      *ASTA  ORB    #$A0      '*' set
0363 5CE7 20 EF      BRA    FSTR1
0364                  *
0365 5CE9 CA 90      *YEN  ORB    #$90      '¥' set
0366 5CEB 20 EB      BRA    FSTR1
0367                  *
0368 5CED CA 08      *PLUS ORB    #8      '+ ' set
0369 5CEF 20 E7      BRA    FSTR1
0370                  *
0371 5CF1 CA 84      *MINUS ORB    #$84      '- ' set
0372 5CF3 20 E3      BRA    FSTR1
0373                  *
0374 5CF5 CA 81      *EXP  ORB    #$81      EXP set
0375 5CF7 20 DF      BRA    FSTR1
0376                  *
0377 5CF9 0D B8      *TEN  TST    <$B8      USING low
0378 5CFB 26 02      BNE    TEN1
0379 5CFD 0C B8      INC    <$B8
0380 5CFF 20 D7      TEN1  BRA    FSTR1
0381                  *
0382 5D01 BD 9732     NEXT1  JSR    $9732      ', ' test
0383 5D04 BD 975F     JSR    $975F      formula
0384 5D07 BD 972C     JSR    $972C      ') ' test
0385 5D0A BD B97A     JSR    $B97A      to ASCII
0386 5D0D 9F 7F      STX    <$7F      STR
0387 5D0F 5F        CLR    B
0388 5D10 6D 80      COUNT  TST    .X+
0389 5D12 27 03      BEQ    CNTEND
0390 5D14 5C        INCB   B
0391 5D15 20 F9      BRA    COUNT
0392                  *
0393 5D17 D7 7E      *CNTEND STB    <$7E      STR
0394 5D19 7E 9C4F    JMP    $9C4F      STR set
0395                  *
0396 5D1C DC C9      *PLOT  LDD    <$C9      end to start
0397 5D1E DD C5      STD    <$C5
0398 5D20 DC CB      LDD    <$CB      end to start
0399 5D22 DD C7      STD    <$C7
0400 5D24 33 8C 68    LEAU   <ABSX,PCR      set ABSX
0401 5D27 108E 00C9  LDY    #$00C9      set point end
0402 5D28 C6 02      LDB    #2
0403 5D2D 34 04      PSHS   B
0404 5D2F CC FCF8     PLOT1  LDD    #$FCF8      set counter
0405 5D32 AE 54      LDX    -12,U      set offset
0406 5D34 2A 02      BPL    PLOT2      load R*U
0407 5D36 1E 89      EXG    A,B
0408 5D38 AE C4      PLOT2  LDX    .U
0409 5D3A AC C5      CMPLX  B,U
0410 5D3C 2C 04      BGE    CHKMAX
0411 5D3E AE C5      LDX    B,U
0412 5D40 20 06      BRA    SETPOT      set R*MIN
0413                  *
0414 5D42 AC C6      *CHKMAX CMPLX  A,U      ABS* : R*MAN
0415 5D44 2F 02      BLE    SETPOT
0416 5D46 AE C6      LDX    A,U      set R*MAX
0417 5D48 AF A4      STX    .Y      set point
0418 5D4A 33 42      LEAU   2,U      set ABSY
0419 5D4C 31 22      LEAY   2,Y      set end Y
0420 5D4E 6A E4      DEC    S
0421 5D50 26 DD      BNE    PLOT1
0422 5D52 32 61      LEAS   1,S      delete work
0423 5D54 0D C4      TST    <PEN
0424 5D56 27 06      BEQ    RET1      check PEN
0425 5D58 7F 01A4    CLR    $01A4      MOV
0426 5D5B 7E E2E5    JMP    $E2E5      LINE
0427                  *
0428 5D5E 39      *RET1  RTS
0429                  *
0430 5D5F          *SFAC  RMB    4      FAC save
0431 5D63          VFX   RMB    4      virtual point X
0432 5D67          VFY   RMB    4      virtual point Y
0433 5D6B          VXMIN RMB    4      virtual min X
0434 5D6F          VYMIN RMB    4      virtual min Y
0435 5D73          VXL   RMB    4      virtual range X
0436 5D77          VYL   RMB    4      virtual range Y
0437 5D7B          BAIX  RMB    4      real/virtual X
0438 5D7F          BAIY  RMB    4      real/virtual Y
0439 5D83 027F      RXL   FDB    $027F      real range X
0440 5D85 00C7      RYL   FDB    $00C7      real range Y
0441 5D87          RXMIN RMB    2      real min X
0442 5D89          RYMIN RMB    2      real min Y
0443 5D8B          RXMAX RMB    2      real max X
0444 5D8D          RYMAX RMB    2      real max Y
0445 5D8F          ABSX  RMB    2      real point X
0446 5D91          ABSY  RMB    2      real point Y
0447                  *
0448 00C4          *PEN  EQU    $C4
0449 5D93          *END  EQU    *
0450                  *

```

```

10 WIDTH 40,20
20 PRINT"RSCALE Xmin,Ymin,Xmax,Ymax"
30 VSCALE 0,0,10,10
40 X=0:Y=0
50 FOR I=1 TO 4
60 X=X+64:Y=Y+20
70 XL=320-X:XH=319+X:YL=100-Y:YH=99+Y
80 RSCALE XL,YL,XH,YH
90 X1=RPOSX(0):Y1=RPOSY(0)
100 A$=FSTR$( "###",XL)+", "+FSTR$(,YL)+", "+FSTR$(,XH)+", "+FSTR$(,YH)
110 SYMBOL (X1+4,Y1+5),A$,1,1
120 MOVA 0,0:DRWA 10,0,PSET,I
130 DRWA 10,10:DRWA 0,10:DRWA 0,0
140 NEXT:LOCATE0,17

```

# R A N D O M B O X

## FM-8 REM オプションAUTO

■井戸隆明

FM-8は、Kコンパイラやアセンブラのソースの作成に、BASICのエディタで、REM文の形で作成します。そのときに使うAUTOコマンドはFM-8の特長である複数行のスクリーン・エディットができませんし、1度リストを出さないと、直すこともできません。今回はこの点を改善し、冒頭に述べたものの作成が容易になるようにFM-11のようなREMオプションをつけ加えたAUTOを作ってみました。

F-BASICのAUTOコマンドは、コマンド待ちループと同じところを回っていて、AUTO FLAG(\$9D)が立っているかないかでその動作の切り換えを行なっています。そこで、キー入力部、中間言語の変換部、リスト挿入部をそっくりそのまま取り出して手を加えたものが、今回のプログラムです。逆アセンブルしたものを参考にして1晩で作りあげたので、かなりムダが多いと思います。

ROM内ルーチンを多用しているのでFM-8しか走りません(Ver.1.05)。

このプログラムはリロケータブルですが、ワーク・エリアを含めて1K強(\$5000~\$5450)のエリアが必要です。スタート番地は\$5000です。また、ALL RAM BASICのようなものに組み込むのであれば、\$5053~\$51D5だけを取り出して\$8FEE、\$8FEF番地にそれぞれロードした番地を入れた後、BASICをSAVEすればいいと思います。

EXEC&H5000とすると“New AUTO ON,/”とメッセージが出て、使える状態になります。使用形式はAUTO([l])(,m) [ ,R]でl,mは本来のものと同じです。Rオプションを付け加えることによって、REM文の“\*”が自動的に出ます。また、動作中にすでに存在している行があった場合“\*”が出て、そこをパスしたいときはReturnを押せば次に進みます。

複数行のスクリーン・エディットはできませんが、文番号なしの部分があるとそこだけ無視します。コマンド待ちに戻るにはSTOP, CTRL+C, CTRL+Xのいずれかで戻ります。

なお、アセンブラなどのオブジェクトをおとすのにこのプログラムが邪魔になりましたら、EXEC(&H5000)と入力し

てください。すると“New AUTO OFF”とメッセージが出て、その領域が開放されます(これをしないと暴走する)。

FM-8は安価なコンパイラが数種類ありますが、これらはBASICの下で動くものばかりで、本格的なOSで動くものと比べると扱いにくいと思います。特にエディタは特筆ものです。

常々、フルスクリーン・エディタ(BASIC形式のもの)を作ろうと思っていましたが、今回はその第1弾のつもりで作りました。

これを使って、どんどんプログラムを作りましょう。

### REM オプションAUTOプログラム

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
5000 8E BF B4 BC 01 F2 26 23 33 BD 01 CA 34 40 EC B1 :35
5010 ED C1 BC 92 2F 25 F7 35 40 FF 01 F2 30 BD 00 33 :4E
5020 AF C8 3A 30 BD 00 0E BD 9F DF 29 BF 01 F2 30 BD :5F
5030 00 11 BD 9F DF 39 4E 65 77 20 41 55 54 4F 20 4F :77
5040 4E 20 21 00 4E 65 77 20 41 55 54 4F 20 4F 46 46 :0D
5050 20 21 00 8E 00 0A 9F 4B 9F 9E B1 2C 27 14 9D DB :5D
5060 27 07 BD A3 6C 9E 4B 9F 9E 8E 00 0A 9F 4B 9D DB :17
5070 27 24 BD 97 32 26 B1 2C 27 0E BD 9E C7 9E 4B 0B :08
5080 9D DB 27 12 BD 97 32 20 02 9D D2 81 52 27 03 7E :40
5090 97 40 0C 9D 9E 4B 9F A0 26 03 7E A3 C0 9E 9E 9F :97
50A0 4B BD DB 3B BD A2 05 BD 93 E2 86 20 10 25 00 02 :61
50B0 86 2A BD D3 52 0D 9D 27 05 B6 27 BD D3 52 17 00 :0E
50C0 C2 B1 02 27 DB 81 01 10 27 00 A9 9F D9 9D D2 10 :9D
50D0 27 00 B3 10 24 00 7F BE FF FF 9F 47 25 0C 96 BF :55
50E0 10 26 41 C5 BD C5 CD 7E C9 C2 BD 96 07 BD D0 BC :07
50F0 DC 4B DD E4 9E 4B BF 03 3A 7F 03 3C 9E D9 30 1F :51

```

Sum: C0 B6 40 B2 49 A6 4F C8 1C 7B 64 C8 DF FE 7A 6A :95

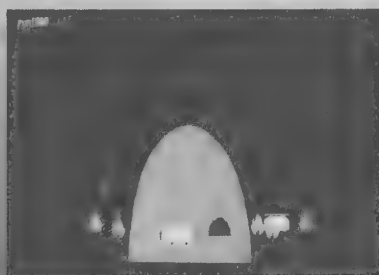
```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
5100 A6 B4 B1 20 27 F8 B1 09 27 F4 A6 01 B1 2A 27 04 :0C
5110 81 20 26 02 30 01 BD C5 C9 DD 13 DC E4 DD 4B BD :DA
5120 93 E2 25 09 9F A2 AE B4 9F A4 BD A3 DF B6 03 3C :BD
5130 27 1E 9E 4B 9F A6 CD 35 DD 61 D3 13 DD 5F BD 92 :33
5140 6D CE 03 3B A6 C0 A7 80 9C 63 26 F8 9E 5F 9F 35 :F1
5150 BD 9A 0E BD CA 66 B6 05 A9 27 06 BD 51 7E 18 FF :7E
5160 6A DC 9E D3 A0 10 25 00 0B B1 F9 10 22 00 05 DD :25
5170 9E 16 FF 29 0F 9D 32 62 BD 9F 97 7E 93 45 CE DC :0F
5180 14 20 0C BD DC FA 73 05 A9 7F 05 B7 CE DC 0C BD :A2
5190 DB FF 1A 40 7F 03 13 9D DE 1C BF BD DC F3 FC 03 :AA
51A0 12 26 07 B6 04 CD 26 07 20 14 B6 01 B7 04 3D 4F :65
51B0 5F B7 05 A9 FD 03 12 BD 9F 97 B6 04 3D 39 EC 04 :E9
51C0 B3 00 04 22 0E 2A 07 B6 05 A9 27 07 20 B0 4F 5F :F4
51D0 B7 05 A9 7E DA A5 00 00 00 00 00 00 00 00 00 :62
51E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
51F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00

```

Sum: AD F9 F7 63 FB 1C 41 8A C4 6F 2C 56 B3 FA 3A EE :39

# 拡張PUT



絵の優先度を  
付けられる  
PUT命令！

■向原 宏

このプログラムはF-BASICのPUT文では不可能だった画面表示を可能にするもので、背景や表示色を乱さずにドット・パターンを表示し、移動させることができます。

なお、プログラムはすべてマシン語で、ポジション・インデペンデントになっています。

## 使 い 方

まず、リスト1, 2, 3を入力してRUNしてみてください。このプログラムがどういう機能を持つか理解できると思います。背景を壊さずに、背景の『こちら側』と『向こう側』にパターン（ここではバス）を表示しているのがわかると思います。このプログラムを使って3次元的なゲームなどに応用すれば、面倒な陰線処理を意識する必要はなくなるでしょう。

実際の使い方はリスト3のデモ・プログラムを見ればわかると思いますが、EXEC & H4C00で背景の『こちら側』に表示され、EXEC & H4C04で背景の『向こう側』に表示されます。その際、背景のデータはVRAMのフリーエリアにストアされるようになっているので、\$180バイトまでのデータを表示することができます。しかし、あまり大きいと表示に時間がかかってしまうので、パタパタはため

いたようになります。デモ・プログラムのバス程度の大きさが適当でしょう。

表示したパターンを消すにはEXEC & H4C08で、元の背景が表示されます。なお、データのアドレスを変更した場合は4C25, 26番地の値も変更してください。

パターンの表示位置は4FFC番地にX, 4FFD番地にYの値をPOKEしてください。ただし、X方向には1ドット単位ではなく、8ドット単位にしか移動しません。そのためXは0~79の値しかとれません（Yは0~199）。

なお、デモ・プログラムをRUNして途中でストップしRUN1000とするとテン・キーでバスを上下左右に動かせます。ただし、画面の範囲外には移動させないでください。

## 最 後 に

アセンブラによる初めての本格的なプログラムなので、無駄な点や非効率なところもありますが、FM-8を手にして10箇月目にしてここまでできたのだから、まあ満足しています。

できれば上間コードを定義してBASICの命令を拡張したかったのですが、まだどうすれば良いかわからず、今回のようにPOKEとEXECで行なわざるをえませんでした。

### リスト1 拡張PUT アセンブル・リスト

```

0001 4C00          DRB  $4C00
0002 4C00 C601    LDB  #1
0003 4C02 2005    BRA  M1
0004 4C04 C6FF    LDB  #$FF
0005 4C06 2001    BRA  M1
0006 4C08 5F      CLR  B
0007 4C09 B6FD05  M1:  LDA  $FD05
0008 4C0C 2BFB    BMI  M1
0009 4C0E 1A50    ORCC #$50
0010 4C10 B680    LDA  #$80
0011 4C12 B7FD05  STA  $FD05
0012 4C15 B6FD05  M2:  LDA  $FD05
0013 4C18 2AFB    BPL  M2
0014 4C1A BE0000  LDX  #0
0015 4C1D BFFCFE  STX  $FCFE
0016 4C20 10BEFC80 LDY  #$FC80
0017 4C24 CE5000  LDU  #$5000
0018 4C27 E75B    STB  -5,U
0019 4C29 2B0A    BMI  M13
0020 4C2B 260E    BNE  M9
0021 4C2D 308D00C8 LEAX TABLE,PC
0022 4C31 C669    LDB  #$69
0023 4C33 200C    BRA  M3
0024 4C35 308D019E M13: LEAX TABLE2,PC

```

```

0025 4C39 2004          BRA  M14
0026 4C3B 308D011D      M9:  LEAX TABLE1,PC
0027 4C3F C67B          M14: LDB  #$7B
0028 4C41 A680          M3:  LDA  ,X+
0029 4C43 A7A0          STA  ,Y+
0030 4C45 5A            DECB B
0031 4C46 26F9          BNE  M3
0032 4C48 7FFD05        CLR  $FD05
0033 4C4B EC5E          LDD  -2,U
0034 4C4D 1F02          TFR  D,Y
0035 4C4F 3422          PSHS A,Y
0036 4C51 E65D          LDB  -3,U
0037 4C53 B650          LDA  #$50
0038 4C55 3D            MUL  B
0039 4C56 EB5C          ADDB -4,U
0040 4C58 B900          ADCA #0
0041 4C5A 1F01          TFR  D,X
0042 4C5C F6FD05        M4:  LDB  $FD05
0043 4C5F 2BFB          BMI  M4
0044 4C61 C680          LDB  #$80
0045 4C63 F7FD05        STB  $FD05
0046 4C66 F6FD05        M5:  LDB  $FD05
0047 4C69 2AFB          BPL  M5
0048 4C6B 10BFFCFE      STY  $FCFE

```

注) 左上の写真ではバスが二台とも「こちら側」にあるが右上のものは右の一台が木にかくれている。

0049	4C6F	BFFCFC		STX	\$FCFC	0131	4D2A	A6C98000	S3:	LDA	\$8000,U
0050	4C72	6D5B		TST	-5,U	0132	4D2E	A7898000		STA	\$8000,X
0051	4C74	2B5F		BMI	M15	0133	4D32	A6C94000		LDA	\$4000,U
0052	4C76	272F		BEQ	M18	0134	4D36	A7894000		STA	\$4000,X
0053	4C78	7FFD05	M16:	CLR	\$FD05	0135	4D3A	A6C0		LDA	,U+
0054	4C7B	1F20		TFR	Y,D	0136	4D3C	A780		STA	,X+
0055	4C7D	3D		MUL		0137	4D3E	0AFC		DEC	<\$FC
0056	4C7E	30CB		LEAX	D,U	0138	4D40	26EB		BNE	S3
0057	4C80	318B		LEAY	D,X	0139	4D42	3510		PULS	X
0058	4C82	A680	M6:	LDA	,X+	0140	4D44	308850		LEAX	<\$50,X
0059	4C84	E6A4		LDB	,Y	0141	4D47	0AFE		DEC	<\$FE
0060	4C86	3406		PSHS	A,B	0142	4D49	2706		BEQ	S4
0061	4C88	AA0A		ORA	,Y+	0143	4D4B	96FD		LDA	<\$FD
0062	4C8A	AA0A		ORA	,U	0144	4D4D	97FC		STA	<\$FC
0063	4C8C	43		COMA		0145	4D4F	20D7		BRA	S0
0064	4C8D	E6C0		LDB	,U+	0146	4D51	C6FF	S4:	LDB	##\$FF
0065	4C8F	3404		PSHS	B	0147	4D53	F7D3FF		STB	\$D3FF
0066	4C91	F6FD05	M7:	LDB	\$FD05	0148	4D56	F6D40A		LDB	\$D40A
0067	4C94	2BFB		BMI	M7	0149	4D59	0F08		CLR	<\$08
0068	4C96	C680		LDB	##80	0150	4D5B	39		RTS	
0069	4C98	F7FD05		STB	\$FD05	0151	4D5C				
0070	4C9B	F6FD05	M8:	LDB	\$FD05	0152	4D5C	00003F			
0071	4C9E	2AFB		BPL	M8	0153	4D5F	59414D4155		TABLE1:	FCB \$00,\$00,\$3F
0072	4CA0	F6FCFF		LDB	\$FCFF			434849		FCC	/YAMAUCHI/
0073	4CA3	270B		BEQ	M10	0154	4D67	93		FCB	\$93
0074	4CA5	2B27		BMI	M12	0155	4D68	D38F		FDB	\$D38F
0075	4CA7	7FFD05	M18:	CLR	\$FD05	0156	4D6A	90		FCB	\$90
0076	4CAA	3476		PSHS	A,B,X,Y,U	0157	4D6B	0A08		DEC	<\$08
0077	4CAC	3576		PULS	A,B,X,Y,U	0158	4D6D	108ED400	S5:	LDY	##\$D400
0078	4CAE	20E1		BRA	M7	0159	4D71	CE3E80		LDU	##\$3E80
0079	4CB0	3504	M10:	PULS	B	0160	4D74	A628		LDA	8,Y
0080	4CB2	FDFCFB		STD	\$FCFB	0161	4D76	A629		LDA	9,Y
0081	4CB5	3506		PULS	A,B	0162	4D78	A62A	S6:	LDA	10,Y
0082	4CB7	FDFCFD		STD	\$FCFD	0163	4D7A	EC3E		LDD	-2,Y
0083	4CBA	7CFCFF		INC	\$FCFF	0164	4D7C	27FA		BEQ	S6
0084	4CBD	7FFD05		CLR	\$FD05	0165	4D7E	E72A		STB	10,Y
0085	4CC0	6AE4		DEC	,S	0166	4D80	AE3C		LDX	-4,Y
0086	4CC2	26BE		BNE	M6	0167	4D82	DDFD		STD	<\$FD
0087	4CC4	6A62		DEC	2,S	0168	4D84	97FC		STA	<\$FC
0088	4CC6	2709		BEQ	M11	0169	4D86	3410	S10:	PSHS	X
0089	4CC8	A661		LDA	1,S	0170	4D88	6F3F	S11:	CLR	-1,Y
0090	4CCA	A7E4		STA	,S	0171	4D8A	A62A	S7:	LDA	10,Y
0091	4CCC	20B4		BRA	M6	0172	4D8C	A63F		LDA	-1,Y
0092	4CCE	7FFD05	M12:	CLR	\$FD05	0173	4D8E	27FA		BEQ	S7
0093	4CD1	1CAF	M11:	ANDC	##AF	0174	4D90	A72A		STA	10,Y
0094	4CD3	35C2		PULS	A,U,PC	0175	4D92	EC3D		LDD	-3,Y
0095	4CD5					0176	4D94	3406		PSHS	A,B
0096	4CD5	308D0012	M15:	LEAX	MDT,PC	0177	4D96	EC3B		LDD	-5,Y
0097	4CD9	C60E		LDB	#14	0178	4D98	97FF		STA	<\$FF
0098	4CDB	10BEFCB0		LDY	##FCB0	0179	4D9A	A684	S8:	LDA	,X
0099	4CDF	A680	M17:	LDA	,X+	0180	4D9C	A7C4		STA	,U
0100	4CE1	A7A0		STA	,Y+	0181	4D9E	94FF		ANDA	<\$FF
0101	4CE3	5A		DECB		0182	4DA0	E784		STB	,X
0102	4CE4	26F9		BNE	M17	0183	4DA2	AAB4		ORA	,X
0103	4CE6	10AE5E		LDY	-2,U	0184	4DA4	A784		STA	,X
0104	4CE9	208D		BRA	M16	0185	4DA6	30894000		LEAX	\$4000,X
0105	4CEB	A684	MDT:	LDA	,X	0186	4DA8	33C94000		LEAU	\$4000,U
0106	4CED	AAB94000		ORA	\$4000,X	0187	4DAE	8CBFFF		CMPL	##BFFF
0107	4CF1	AAB98000		ORA	\$8000,X	0188	4DB1	2A04		BPL	S20
0108	4CF5	43		COMA		0189	4DB3	3504		PULS	B
0109	4CF6	97FF		STA	<\$FF	0190	4DB5	20E3		BRA	S8
0110	4CF8	39		RTS		0191	4DB7	30894001	S20:	LEAX	\$4001,X
0111	4CF9					0192	4DBB	33C94001		LEAU	\$4001,U
0112	4CF9					0193	4DBF	0AFC		DEC	<\$FC
0113	4CF9	00003F				0194	4DC1	26C5		BNE	S11
0114	4CFC	59414D4155	TABLE:	FCB	\$00,\$00,\$3F	0195	4DC3	3510		PULS	X
		434849		FCC	/YAMAUCHI/	0196	4DC5	308850		LEAX	<\$50,X
0115	4D04	93		FCB	\$93	0197	4DC8	0AFE		DEC	<\$FE
0116	4D05	D38F		FDB	\$D38F	0198	4DCA	2706		BEQ	S9
0117	4D07	90		FCB	\$90	0199	4DCC	96FD		LDA	<\$FD
0118	4D08	0A08		DEC	<\$08	0200	4DCE	97FC		STA	<\$FC
0119	4D0A	B6D408		LDA	\$D408	0201	4DD0	20B4		BRA	S10
0120	4D0D	B6D409		LDA	\$D409	0202	4DD2	A62A	S9:	LDA	10,Y
0121	4D10	CE3E80		LDU	##3E80	0203	4DD4	0F08		CLR	<\$08
0122	4D13	B6D40A	S2:	LDA	\$D40A	0204	4DD6	39		RTS	
0123	4D16	B6D3FE		LDA	\$D3FE	0205	4DD7				
0124	4D19	27F8		BEQ	S2	0206	4DD7	00003F		TABLE2:	FCB \$00,\$00,\$3F
0125	4D1B	F7D40A		STB	\$D40A	0207	4DDA	59414D4155		FCC	/YAMAUCHI/
0126	4D1E	F6D3FF		LDB	\$D3FF			434849			
0127	4D21	BD03FC		LDX	\$D3FC	0208	4DE2	93		FCB	\$93
0128	4D24	DDFD		STD	<\$FD	0209	4DE3	D38F		FDB	\$D38F
0129	4D26	97FC		STA	<\$FC	0210	4DE5	90		FCB	\$90
0130	4D28	3410	S0:	PSHS	X	0211	4DE6	0C08		INC	<\$08

## リスト1 拡張PUT アセンブル・リスト

```

0212 4DE8 10ED400 LDY #D400
0213 4DEC CE3E80 LDU #3E80
0214 4DEF A628 LDA 8,Y
0215 4DF1 A629 LDA 9,Y
0216 4DF3 A62A S12: LDA 10,Y
0217 4DF5 EC3E LDD -2,Y
0218 4DF7 27FA BEQ S12
0219 4DF9 E72A STB 10,Y
0220 4DFB AE3C LDX -4,Y
0221 4DFD DDFD STD <#FD
0222 4DFF 97FC STA <#FC
0223 4E01 3410 S13: PSHS X
0224 4E03 6F3F S14: CLR -1,Y
0225 4E05 BDD380 JSR $D380
0226 4E08 A62A S15: LDA 10,Y
0227 4E0A A63F LDA -1,Y
0228 4E0C 27FA BEQ S15
0229 4E0E A72A STA 10,Y
0230 4E10 EC3D LDD -3,Y
0231 4E12 3406 PSHS A,B
0232 4E14 E63C LDB -4,Y
0233 4E16 D4FF S16: ANDB <#FF
0234 4E18 A684 LDA ,X
0235 4E1A A7C4 STA ,U
0236 4E1C EAB4 ORB ,X
0237 4E1E E784 STB ,X
0238 4E20 30B94000 LEAX $4000,X
0239 4E24 33C94000 LEAU $4000,U
0240 4E28 BCBFFF CMPLX #BFFF
0241 4E2B 2A04 BPL S17
0242 4E2D 3504 PULS B
0243 4E2F 20E5 BRA S16
0244 4E31 30B94001 S17: LEAX $4001,X
0245 4E35 33C94001 LEAU $4001,U
0246 4E39 0AFC DEC <#FC
0247 4E3B 26C6 BNE S14
0248 4E3D 3510 PULS X
0249 4E3F 30B850 LEAX <#50,X
0250 4E42 0AFE DEC <#FE
0251 4E44 2706 BEQ S18
0252 4E46 96FD LDA <#FD
0253 4E48 97FC STA <#FC
0254 4E4A 20B5 BRA S13
0255 4E4C A62A S18: LDA 10,Y
0256 4E4E 0F08 CLR <#08
0257 4E50 39 RTS

```

## リスト2 デモ用データ ダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4EE0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
4EF0	00	00	00	00	00	00	00	00	00	00	00	01	32	91	03	13	1DA
Sum:	00	00	00	00	00	00	00	00	00	00	00	01	32	91	03	13	1DA

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4F00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
4F10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
4F20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
4F30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
4F40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
4F50	00	00	00	00	00	00	00	00	04	00	00	0E	40	01	CE	C0	1E1
4F60	00	7F	80	00	1F	00	00	1E	00	00	0E	00	00	0E	00	00	158
4F70	0E	00	00	0C	00	00	1E	00	00	3F	80	04	3F	C0	04	7F	17D
4F80	E0	0C	FF	E0	0F	FF	F0	0F	FF	FB	1F	FF	FD	3F	FF	FF	127
4F90	1F	FB	FF	1F	F1	BE	0E	31	3C	07	80	78	01	C0	E0	00	102
4FA0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
4FB0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
4FC0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
4FD0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
4FE0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100
4FF0	00	00	00	00	00	00	00	00	00	00	00	FF	40	95	05	0E	1E7
Sum:	0D	B6	7E	0B	1F	BD	1C	5E	3F	3E	2D	BB	BD	63	B6	4C	C6

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
5000	7F	FF	FF	FF	FE	00	30	00	0C	00	03	FF	FF	FF	FF	C0	1F	195
5010	FF	FF	FF	FF	7F	FF	FF	FF	FE	FF	FF	FF	FF	FF	FF	FF	FC	162
5020	FC	FC	FF	FC	FC	FC	FC	FF	FC	FC	FC	FC	FF	FF	FF	FF	FF	D2
5030	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	00	1F3	
5040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
5050	03	FF	FF	FF	C0	1F	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	14D	
5060	FF	FF	FF	FF	FF	FF	FF	FF	E7	FF	FF	FF	FF	FF	E7	FF	1C0	
5070	FF	FF	E7	FF	FF	FF	FF	E7	FF	FF	FF	FF	FF	FF	C0	00	183	
5080	00	03	07	E0	FF	07	E0	03	C0	00	03	C0	00	00	00	00	156	
5090	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
50A0	00	00	00	00	00	07	87	87	87	80	07	87	87	87	98	07	157	
50B0	87	87	87	98	07	87	87	87	98	07	87	87	87	98	00	00	195	
50C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
50D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
50E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
50F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	100	
Sum:	01	80	6F	67	3C	AC	15	F3	C9	77	0B	C4	0B	C2	4F	1F	18E	

## リスト3 デモ用BASICプログラム・リスト

```

10 CLS:CIRCLE(300,180),101,1,.99,.5,1,F:CIRCLE(350,160),20,2,.7,.5,1,F:CIRCLE(35
0,160),17,0,.7,.5,1,F
20 X=10:Y=150:AD=&H4C00:AD1=AD:DX=1
25 POKE AD+&H25,&H4F:FOR I=0 TO 5:READ TX,TY:POKE &H4F00-4,TX:POKE &H4F00-3,TY:E
XEC AD:NEXT:POKE AD+&H25,&H50
30 POKE &H5000-4,X:POKE &H5000-3,Y:EXEC AD+2-2*DX
35 FOR I=0 TO 100:NEXT
40 IF X=65 THEN DX=-1:GOTO90
50 IF X=10 THEN DX=1:GOTO90
60 IF X<15 THEN DY=1:GOTO90
70 IF X>60 THEN DY=-1:GOTO90
80 DY=0
90 X=X+DX:Y=Y+DY
100 EXEC AD+8:GOTO30
110 DATA 16,144,21,148,30,146,60,143,53,148,50,145
1000 REM *****
1010 X=30:Y=100:AD=&H4C00:AD1=AD:GOTO 1110
1020 A$=INKEY$:IF A$<>" " THEN B=VAL(A$)
1030 ON B GOTO 1060,1070,1080,1050,1090
1040 IF A$="+" THEN AD=AD1 ELSE IF A$="-" THEN AD=AD1+4
1050 GOTO 1020
1060 X=X-1:GOTO 1100
1070 Y=Y+3:GOTO 1100
1080 X=X+1:GOTO 1100
1090 Y=Y-3
1100 EXEC AD1+8
1110 POKE &H5000-4,X:POKE &H5000-3,Y:EXEC AD
1120 FOR I=0 TO 100:NEXT
1130 GOTO 1020

```



# FM-8

## バージョン・アップが簡単にできる

# DOS モード (ALL RAM) DISK BASIC

横須賀マイコンクラブ 橋口義人

FM-8は、FLEX、CP/MなどいろいろなDOSを走らせることができる、すぐれたマイコンだと思います。しかしDISK BASICと他のDOSを使う場合、モードの切り換えを後面のディップ・スイッチにより行なわなければなりません。この煩わしさを解消するために、DOSモードでDISK BASICを走らせることができないかを見当ててみました。

CP/MなどのDOSは、DOSそのものをディスケットに書き込んであり、BOOTプログラムでRAMに読み込んで、システムを起動しています。そこでDISK BASICでも、同じようにインタープリタをディスケットに書き込んで、BOOTプログラムを作り、DOSモードで起動できるようになりました。

これにより、DOSの違いによりディップ・スイッチを変更することなく、起動できるようになりました。なおALL RAMのためインタープリタのバージョン・アップすることも簡単にできるようになりました。

## DOSモードのシステム・ディスケットの作り方

まず、普通にDISK BASICを起動します。システム・ディスケットについている“SYSDSK”をRUNし、ブランク・ディスケットをフォーマットし、システムをコピーします。

この後にリスト1のBOOTプログラムを入力します。入力が終わったらDISKにセーブしておきます。このプログラムでは、マシン語をDATA文にしてあるので、ミスのないように入力してください。

RUNして、もしDATA文のところでミスがあれば、プログラムはミスのあった行番号を表示してストップするので、もういちどリストを見て、確認してください。

ミスがなければ、しばらくして“Program End”と表示してDOSモードのシステム・ディスケットができあがります。

## DOSモードBASICの起動方法

DOSモード・ディスケットをドライブ0に入れ、後面のディップ・スイッチをDOSモードに設定してリセット・ボタンを押します。するとしばらくして、従来通りのDISK BASICが走りだします。

以上の操作の画面のハード・コピーを図1に示します。

## バージョン・アップの方法

このときすでにRAM上でBASICが動いているので、モニタなどにより、パッチを行いません。

その後、SAVEM“FBAS. SYS”, &H8000, &HFBFF, 6809 [CR]とすると、バージョン・アップされたDISK BASICができあがります。

F-BASICの場合、8000番地から87FF番地あたりまで空いているので、そこにプログラムを追加できます。私の場合はモニタもバージョン・アップしてこのエリアに追加しています。

## DOSモード・ディスケットのバック・アップの方法

オリジナルのシステム・ディスケットの“VOLCOPY”プログラムを使って全トラック・コピーして後で不要のファイルを消す方法がありますが、面倒です。

そこで“SYSGEN”なるプログラムに“SYSDSK”プログラムを変更しました。リスト2に示すところをなおしてください。このプログラムはオリジナルのシステム・ディスケットのマシン語プログラム“DFMCD”を使うので、SYSGENプログラムをRUNする前に、LOADM“DFMCD”を実行し、あらかじめRAMに読み込んでおきます。

その後RUNするとブランク・ディスケットをフォーマットし、DOSモードBASIC、SYSGENおよびDFMCDを作ります。

## その他

リセット後DOSモードで起動したあとは、いままでのディスケットでも自由に読み書きできます。DOSモードBASICのディスケット、“FBAS. SYS”のファイルは、FATの最初の連続したクラスタにない起動しないので、バージョン・アップのときはファイルが最初にあることを確認してから行なってください。普通BOOTプログラムかSYSGENプログラムで作ったものはファイルの最初に作られています。



## 終わりに

このプログラムがF-BASICのバージョン・アップに役に立てば幸いに思います。

### 参考文献

- 1) I/O FM-8 に関する号
- 2) FM-8 活用研究
- 3) "富士通FM-8マニュアル"

図1 操作手順

<pre> DISK BASICでシステム起動 DISK VERSION How many disk drives(1-4)? How many disk files(0-15)?  FUJITSU MICRO B BASIC Version 1.0 Copyright (C) 1981 By FUJITSU/MICROSOFT 25603 Bytes Free  poke %hfd00,%h40 RUN"SYSDSK"  *** FLOPPY DISK FORMATTING ***  DISK FORMATTING ? Y WHAT DISK DRIVE ? 1 SET THE FORMATTING DISK ON DRIVE 1 READY ? Y DISK FORMATTING COMPLETED !! *** DISK BASIC SYSTEM COPY ***  1. DRIVE 0 --&gt; 1 2. DRIVE 0 --&gt; 0 = 1 READY ? Y SYSTEM CODE COPY COMPLETED !!  2. この時点でリスト1のBOOTプログラムを入力し、ディスクにセーブします。 </pre>	<pre> RUN"BOOT" ** DOS Mode (All RAM) Disk BASIC BOOT Program **  Copyright 1982 (C) By Hashiguchi Soft.  What Drive No ( 0 / 1 ) ? 1   ドライブ番号を指定  +++++  ディスクの"シ"ュンビ" OK ? ON Drive # 1 ( Y/N ) ? Y FBAS.SYS  2 B S 16 (ファイル表示) 136 Clusters Free Program End  3. ここで、ドライブ1にDOSモードDISK BASICができています。 それをドライブ0に移し、後面のディップ・スイッチをDOSモード に設定してリセット・ボタンを押します。 DISK VERSION How many disk drives(1-4)? How many disk files(0-15)?  FUJITSU MICRO B BASIC Version 1.0 Copyright (C) 1981 By FUJITSU/MICROSOFT 25603 Bytes Free  4. タイトルが表示され、DOSモードでシステムが 起動しました。 READY poke %hfd00,%h40  Ready HARDC </pre>
---	--

リスト1 DISK BASIC BOOT プログラム・リスト

```

10 CLEAR300,%H2FFF
20 "Date 1982-9-24
30 M1$="** DOS Mode (All RAM) Disk BASIC BOOT Program **"
40 M2$=" Copyright 1982 (C) By Hashiguchi Soft."
50 WIDTH80,25:PRINTM1$:PRINTM2$:PRINT
60 PRINTCHR$(7);:INPUT "What Drive No ( 0 / 1 )";D
70 IF D<0 OR D>1 THEN PRINTCHR$(7);:GOTO 50
80 PRINT:FOR I=0 TO 16:PRINT"+":
90 SM=0:READ DT$:FOR J=0 TO 15:DT=VAL("&H"+(MID$(DT$, (J*2+1), 2)))
100 POKE %H30F0+I*16+J,DT:SM=SM+DT:NEXT J
110 READ:CS:IF CS<>SM THEN 250
120 NEXT I
130 PRINT:POKE %H30FF,D
140 PRINT:PRINTCHR$(7);;"ディスクの"シ"ュンビ" OK ? ON Drive #";D;" ( Y/N )";
150 INPUT A$
160 IF A$<>"Y" THEN 140
170 DSKINI D
180 D$="";IF D=1 THEN D$="1:"
190 F$=D$+"FBAS.SYS":SAVEM F$,%H8000,%HFBFF,6809
200 EXEC %H30F0
210 PRINTCHR$(7);:IF PEEK(%H30F9)<>0 THEN PRINT"Disk Write Error":GOTO 240
220 FILES D$
230 PRINT"Program End"
240 CLEAR300,%H5FFF:END
250 PRINT:PRINT:PRINT "DATA フ"ン ニュウリョク ERROR in #";1000+I*10:CHR$(7):GOTO 240
1000 DATA BE30F8BDF2D8390009003100000010000,1201
1010 DATA 328CFD86FD1F8B960485021027008020,1504
1020 DATA 180A007FFB0201000008000000000000,423
1030 DATA 000A007000000010100B67DC60A308CE1,1004
1040 DATA 3402BDFE0B3502252D4A274D340210AE,1076
1050 DATA 02CC010031AB10AF023502E6055CE705,1238
1060 DATA C1112BD7C601E705E604EB06E704E606,1849
1070 DATA C801E70620C53402A601B10C270EB10D,1224
1080 DATA 270A810F270686B1970320FE308C9ABD,1472
1090 DATA FE0235025A26A620ED308C8BDFE0286,17B2
1100 DATA 10C60A308C8B3402BDFE0B350225124A,1240
1110 DATA 27373402CC0100E302ED026C05350220,1021
1120 DATA E03402A601B10C2708B10D2704810F26,1000
1130 DATA B530BDF54BDFE0235025A26C68629B7,1893
1140 DATA FD034F1F8B6E9FFBFE4F1F8B8E700086,1916
1150 DATA FF6E9FFBFE436F707972696768742031,2063
1160 DATA 39383220B432920427920422E4D2E53,912

```

```

550 DSKINI DRV2
560 IF DRV2=0 THEN DR$="0:" ELSE DR$="1:"
570 F1$=DR$+"FBAS.SYS":F2$=DR$+"SYSGEN":F3$=DR$+"DFMCD"
580 SAVEM F1$,&H8000,&HFBFF,&H890A
590 SAVE F2$,P:SAVEM F3$,&H5000,&H512F,&H5000
600 COLOR 6:PRINT "SYSTEM CODE COPY COMPLETED !!":CHR$(7)
610 FILES DR$:COLOR 7:END

```

## R A N D O M B O X

### Display Sub Systemの使い方

ディスプレイ・サブシステムは128バイトの共有RAMを使ってやり取りをします。このとき必ず、サブCPUがREADYになっているときにサブCPUをHALTしなければなりません。このとき割り込みは禁止します。そしてコマンド・コードなどを共有RAMに書いて、サブCPUのHALTを解除します。

この繰り返しでどんどんサブCPUへコマンドなどを送ります。このとき続けてサブCPUにコマンドなどを送る場合は、タイマを入れてください。そうしないと動作がおかしくなります。

また、コマンド、パラメータの間違いでも動作がおかしくなります。特にグラフィック・コマンドのときはデータ量が多いので気を付けてください。

それではコンソール・コマンドとグラフィック・コマンドの例を載せておきますので、実験してみてください。

コンソール・コマンドはINITコマンドです。DATAは80×25文字、ファンクション・キー表示、画面消去、単色表示です。また、グラフィックのほうは例1のDATAに置き換えるものです。このデータによると「ASDCB」が縦、横5倍の大きさで表示されます。つまりSYMBOL文と同じです。

みなさんもFM-8活用研究やFM-7のシステム仕様書を見ながらいろいろ試してみてください。

#### 例1 コンソール・コマンド実行例

0001	0000	HLT=	\$FD05		
0002	0000	BUF=	\$FC82		
0003	0000	:			
0004	6000	:	ORG	\$6000	
0005	6000	:			
0006	6000	B6FD05	START:	LDA	HLT
0007	6005	28FB		RMI	START
0008	6005	1A50		ORCC	#\$50
0009	6007	B680		LDA	#\$80
0010	6009	B7FD05		STA	HLT
0011	600C	30B00012		LEAX	DATA,PC
0012	6010	31B09C6E		LEAY	BUF,PC
0013	6014	EC90	LOOP:	LDD	.Y+
0014	6016	B1FF		CHFA	#\$FF
0015	6018	2704		REQ	END
0016	601A	EDA0		STD	.Y+
0017	601C	20F6		BRA	LOOP
0018	601E	7FFD05	END:	CLR	HLT
0019	6021	39		RTS	
0020	6022	01	DATA:	FCB	1
0021	6023	00		FCB	0
0022	6024	50		FCB	80
0023	6025	19		FCB	25
0024	6026	00		FCB	0
0025	6027	17		FCB	23
0026	6028	01		FCB	1
0027	6029	00		FCB	0
0028	602A	01		FCB	1
0029	602B	FF		FCB	\$FF

;SubCPU Busv?  
 ;  
 ;SubCPU Halt & IRQ Mask  
 ;  
 ;Xreq=DATA Address  
 ;Yreq=\$FC82 コマンド" ラカクアドレス  
 ;Dreq=DATA  
 ;If DATA=\$FF THEN GOTO END  
 ;  
 ;BUF に DATA テンソウ  
 ;GOTO LOOP  
 ;SubCPU Halt カイシ  
 ;モトノルーション カイシ  
 ;コマンド" コート"  
 ;バックカラー  
 ;クリアウ  
 ;バックウ  
 ;スクロール カイシ オウ  
 ;スクロール フル オウ スウ  
 ;ファンクションキー ヒョウシ  
 ;カ"メン ショウキョ (ケサナイ ニ セツテイ)  
 ;グリーン テ" ヒョウシ (フル)  
 ;END Code

#### 例2 グラフィック・コマンド実行例

0020	6022	19	DATA:	FCB	\$19	;コマンド" コート"
0021	6023	03		FCB	3	;カラー コート"
0022	6024	00		FCB	0	;ファンクション コート"
0023	6025	02		FCB	2	;アングル コート"
0024	6026	05		FCB	5	;ヨコ ハ"イリツ
0025	6027	05		FCB	5	;タテ ハ"イリツ
0026	6028	0000		FDB	0	;X サ"ヒョウ
0027	602A	0064		FDB	100	;Y サ"ヒョウ
0028	602C	05		FCB	5	;モシ" スウ
0029	602D	4153444342		FCC	/ASDCB/	
0030	6032	FF		FCB	\$FF	;END Code

長いプログラムをディスクにセーブ!

## DISK SAVER &amp; LOADER

ディスク・セーブ &amp; ロード

■中村俊之

FMのディスク・モードではROMモードに比べて大幅にフリーエリアが減少します。そのためROMモードでは動いていたプログラムが、ディスク・モードではout of Memoryとなり、実行どころかテープからロードすることさえできなくなるものがあります。したがってディスクへのセーブもできません。そのため、このようなプログラムを使うときは、毎回ROMモードでテープからロードする必要があり、ディスクの恩恵にあずかることができません。そこで、そのわずらわしさから解放されるためにこのプログラムを作りました。



## プログラムの内容

F-BASICは、そのI/OがBIOSという非常に使いやすい構造になっているので、I/O関係のルーチンをマクロ命令的に使えます。また、当然のことかもしれませんがディスク・アクセスのサブルーチンが、ROMモードでも存在しているのでマシン語レベルでは、ROMモードであってもディスクを自由に使えます。

さてその方法ですが、ROMモードとディスク・モードの違いがRAM上にディスク・コードおよびディスク用のバッファがあるかないかという点を利用しています。実際に行なっていることは次のとおりです。まず、ROMモードでRAM(\$0000~\$7FFF)の内容(もちろんディスク・コードはなくプログラムが詰まっている)をディスクにセーブしておきます。次に、必要に応じてそれをRAM上にロードすれば、セーブしたときと同じ状態になります。もちろんワーク・エリアもセーブ、ロードするわけですから、完全に元と同じ状態です(図1)。

プログラムについてですが、ディスクに対する入出力はBIOSにまかせっきりなので簡単ですが、その前にFAT、ディレクトリを調べて入出力の対象となるクラスタを見つける必要があります。セーブの場合はまずFATを読んでフリークラスタ(\$FF)を調べ、16クラスタ分(\$0000~\$7FFFをセーブするのに必要なクラスタ数)を確保します。FATを書き直した後、ディレクトリを読んできて、フリーエリア(先頭が\$00か\$FF)にファイル名を記入します。このときプログラムを256バイトに収めるために、正式の名称はNAMEで付け換えることにして、仮の名前としてASCIIコードの\$FE、すなわち“■”を付けています。この文字を使った理由はこのキャラクタがキーボードから

入力できないからです。というのは、このプログラムで作られたファイルは機械語ファイルに近いものですが、ファイルの属性を単純にBASICファイルの形にしているため、誤ってLOADなどの命令で誤動作しないための用心です(もしキャラクタを変更したいなら、リスト1の方は1080行めの6番め、リスト2の方は1020行めの5番めのFEを他のものに変えてください)。最後に確保したフリークラスタに\$0000~\$7FFFのRAMの内容をセーブします。

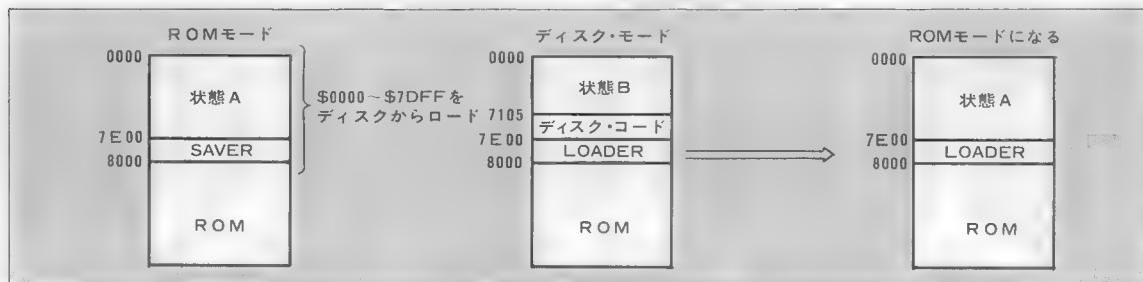
ロードの場合も基本的には同じです。まずディレクトリを読んできてファイルを探します(この場合もファイル名の先頭が“■”のものだけ調べるようにしています)。次にFATを調べてファイルの使っているクラスタを見つけてワーク・エリアに記入しておきます。そして最後に目的のクラスタから\$0000~\$7DFFの分の内容をRAM上にロードし、その後BASICのホット・スタートヘジャンプします。ここで\$7DFFまでしかロードしていないのは、これ以後にある本体を壊さないためです。

前にも述べたように、このプログラムはセーブ、ロードとも本体+ワーク・エリアを256バイト以内、ディスク、I/Oバッファとして256バイト、計512バイトを\$7E00~\$7FFFに置いています。したがって\$7DFFを越えないプログラムについては完全に使えます。もちろん、ポジション・インデペンデントなのでどこに置いても使えますが、あまり意味はないと思います。

ロード用のプログラムですが\$7E00~\$7FFFを使うので、ディスク・モードで使うとディスク・コードを壊してしまいます。うまくロードできた場合は、ROMモードになるのでかまわないのですが、エラーを生じたとき、そのままにしておくとその後、ディスク関係の命令がうまく動かなくなる(実際はエラーメッセージを出す段階で暴走することが多い)ので次のようにしています。初めに本体を\$7000~\$70FFに置いて\$7E00~\$7FFFのディスク・コードを\$6E00~\$6FFFに移します。次に本体を\$7E00以後に移し、メイン・プログラムにジャンプします。エラーを生じたときは、元のプログラムに移ってディスク・コードを元に戻し、BASICのエラーメッセージ・ルーチンヘジャンプします。以上の操作によりディスク・コードの破壊を防いでいます。

またエラーメッセージについての注意ですが、プログラムを256バイト以内にまとめたかったため、BIOSエラーを判定していません。BIOSエラーについてはすべて、エラーコード21(Unprintable error)で出すようにしています。

図1 メモリ・マップによる状態の変化



注) ディスクがnot Readyなどのときは、BIOS中から直接エラーメッセージ・ルーチンへ飛ぶ場合があります。このときはもちろん、ディスク・コードが復帰されないで暴走します。この場合はリセットするしかありません。また、セーブのときはディレクトリFATを直接変更するので、慣れないうちは壊してもよいディスクで練習してみてください。



## 使用法1 (セーブの場合)

まずROMモードで(このプログラムはディスク・モードで使っても意味がありません)、セーブ用のプログラム、(リスト2)を入力して実行します。もちろんその前にテープにセーブしておくのは当然です。そうすると“Drive No=?”と尋ねてくるので、セーブするディスクの入っているドライブ番号を入力します(0または1:4ドライブの方は140行目を修正してください)。

その後、本体を\$7E00以後に設定し(マシン語のデータが誤まっているときは、その行を表示して終わります)、Readyとなるので目的のプログラムをテープからロードします。このときOut of Memoryになった場合は、文字領域をできるだけ小さくしてみてください。それでもだめな場合は、プログラム自身が\$7E00以上になる場合なので、セーブは不可能です。

ロードが終わった後はEXEC&H7E00を実行すると、ディスクにRAM上の内容(ROMモードのプログラム)がセーブされます(約10秒)。もちろんこのときドライブには、16クラス以上のフリークラスをもつディスクを入れておいてください。

次にディスク・モードに移り(リセットまたはEXEC&HFE00)、FILESによってディスクのファイル調べます。“■”という名のファイルがあるはずですから、NAME文によってファイル名を“■×××××××”の形に変えておきます。この処理はロード用のプログラムにとっては必要ありませんが、ユーザーの識別のために行ないます。

セーブのプログラムでの注意ですが、RCBの初期設定をプログラム中で行なっていません(マシン語のデータ中で既にセットしている)。そのためエラーがでてリトライする場合は、もう一度始めから行なうか、もしくはRCBの部分、\$7EE7以後を08, 00, 7F, 00, 01, 01, 00, 0? (ドライブNo)と書き換えて、EXEC&H7E00を実行してください。



## 使用法2 (ロードの場合)

ディスク・モードでロード用のプログラム(リスト1)を入力し、ディスクにセーブしておきます。実行すると“Drive No.=?, File No.=?”と聞いてくるので、それぞれ入力してください(マシン語データにエラーがあった場合はセーブの場合と同様に、その行を表示して終了します)。File No.は、先頭が“■”となっているファイルの初めから何番めかという番号です。

ファイルが存在しない場合には、“File not Found”を表示してコマンド・レベルに戻るので、ディスク内のファイルを確認してください。うまくいった場合は10数秒でBreak/Readyとなり、ROMモードになっています。

その後はCLEAR,&H7FFFの後、普通どおり実行できます。もちろんこの場合はROMモードになっているわけなので、ディスク・コマンドは使えません。



## 最後に

このプログラムではBIOSエントリ・アドレス、BASICホット・スタート・アドレスをエクステンド・インダイレクトで呼んでいる(\$FBFA以後にBIOSエントリ、BASICホット・スタート、BASICコールド・スタートのアドレスが入っている)ので、エラーメッセージ・ルーチンのアドレスを\$8DD1に変更するだけで、FM-7でもそのまま使うことができます。

またディスク・モードからROMモードへの転換をする場合、EXEC&H890A、もしくはモニタからG890A(BASICコールド・スタート)を行なうとあっさり変わってきます。この場合リセットされないで、TIMESなどはそのまま継続されます。ご存じない方はやってみてください。意外と便利です。

なにぶん文章が下手なので、わかりにくいところも多いと思いますが、試行錯誤でやってみてください。

最後にFM-7を気楽に使わせていただいた久留米カホバーの皆さまに誌上を借りてお礼申しあげます。

### 参考文献

- 1) F-BASIC 文法書、富士通
- 2) FM-8活用研究、工学社
- 3) インターフェイス'82年6月号、CQ出版

### リスト1 SAVER

```

1  '
2  '   Saver of the Program under ROM-mode Ver 1.1   :   Date 83/01/31
3  '
100 CLEAR ,&H7DFF
110 DEFINT A-Z

```

```

120 ER=0
130 PRINT:INPUT "Drive No.=";DRV
140 IF DRV<0 OR DRV>1 THEN 130
150 FOR AD=&H7E00 TO &H7EFO STEP 16
160 : TTAL=0
170 : FOR I=0 TO 15
180 : : READ DT$:DT=VAL("&H"+DT$)
190 : : POKE AD+I,DT
200 : : TTAL=TTAL+DT
210 : NEXT I
220 : READ LIN$,SUM$
230 : IF VAL("&H"+SUM$)>>TTAL THEN BEEP:ER=-1:PRINT "Error Line = ";LIN$
240 NEXT AD
250 IF ER=-1 THEN END
260 POKE &H7EEE,DRV
270 END
280
1000 DATA 34,7F,30,8D,00,E1,AD,FF,FB,FA,10,25,00,D3,86,0A, 1000,78A
1010 DATA A7,84,AD,FF,FB,FA,10,25,00,C7,10,AE,02,31,25,30, 1010,70E
1020 DATA 08,1F,13,C6,FF,4F,5C,C1,98,10,24,00,B4,A6,AB,81, 1020,6BD
1030 DATA FF,26,F2,4F,E7,C0,E7,84,30,AB,6A,8D,00,AB,26,E5, 1030,8FD
1040 DATA C6,C7,E7,84,30,8D,00,9F,6A,84,AD,FF,FB,FA,10,25, 1040,91B
1050 DATA 00,8F,86,04,6C,84,A7,05,AD,FF,FB,FA,10,25,00,B1, 1050,70C
1060 DATA B6,B0,EE,02,33,C9,00,80,E6,C6,27,16,C1,FF,27,12, 1060,754
1070 DATA BB,20,28,F4,A6,05,4C,81,11,25,DB,CC,01,01,E7,06, 1070,60B
1080 DATA 20,D4,33,C6,CC,FE,20,ED,C1,86,20,ED,C1,ED,C1,ED, 1080,A74
1090 DATA C1,86,18,6F,C0,4A,26,FB,A6,0B,A7,C8,EE,6A,84,AD, 1090,89F
1100 DATA FF,FB,FA,25,3C,10,8E,00,00,33,0B,A6,C0,1F,89,C4, 1100,700
1110 DATA 01,58,58,58,5C,E7,05,44,1F,89,C4,01,E7,06,44,8B, 1110,5BE
1120 DATA 02,A7,04,C6,0B,10,AF,02,AD,FF,FB,FA,25,13,31,A9, 1120,6EF
1130 DATA 01,00,10,8C,80,00,24,07,6C,05,5A,26,EB,20,CC,35, 1130,442
1140 DATA FF,C6,15,7E,92,AB,10,0B,00,7F,00,01,01,00,00,00, 1140,42E
1150 DATA 00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00, 1150,000

```

## リスト2 LOADER

```

1
2 Loader of the Program under ROM-mode Ver 1.3 : Date 83/01/31
3
100 CLEAR ,&H6DFF
110 DEFINIT A-Z
120 ER=0
130 FOR AD=&H7000 TO &H7130 STEP 16
140 : TTAL=0
150 : FOR I=0 TO 15
160 : : READ DT$:DT=VAL("&H"+DT$)
170 : : POKE AD+I,DT
180 : : TTAL=TTAL+DT
190 : NEXT I
200 : READ LIN$,SUM$
210 : IF VAL("&H"+SUM$)>>TTAL THEN BEEP:ER=-1:PRINT "Error Line = ";LIN$
220 NEXT AD
230 IF ER=-1 THEN END
240
250 PRINT
260 INPUT "Drive No.=";DRV
270 IF DRV<0 OR DRV>1 THEN 260
280 INPUT "File No.=";FIL
290 POKE &H70C7,DRV
300 POKE &H70CB,FIL
310 EXEC &H7100
320
1000 DATA 10,EF,8D,00,B7,10,CE,FC,7F,30,8D,00,B3,86,04,A7, 1000,73D
1010 DATA 05,AD,FF,FB,FA,10,25,00,9C,86,80,EE,02,33,C9,00, 1010,769
1020 DATA 80,E6,C6,C1,FE,26,06,6A,8D,00,9D,27,18,6A,8D,00, 1020,6E1
1030 DATA 8D,27,7E,8B,20,2B,EA,A6,05,4C,81,11,25,D1,CC,01, 1030,63B
1040 DATA 01,E7,06,20,CA,33,C6,E6,4E,31,8D,00,7C,E7,A0,86, 1040,74C
1050 DATA 0F,A7,8C,6B,4F,CE,01,00,EF,05,AD,FF,FB,FA,25,55, 1050,7DA
1060 DATA EE,02,33,45,E6,CB,C1,98,24,4B,E7,A0,6A,8C,50,26, 1060,7D4
1070 DATA F3,10,8E,00,00,33,8D,00,50,A6,C0,1F,89,C4,01,58, 1070,5CC
1080 DATA 58,58,5C,E7,05,44,1F,89,C4,01,E7,06,44,8B,02,A7, 1080,60E
1090 DATA 04,C6,0B,10,AF,02,AD,FF,FB,FA,25,19,31,A9,01,00, 1090,64D
1100 DATA 10,8C,7E,00,02,07,6C,05,5A,26,EB,20,CC,6E,FF,FB, 1100,672
1110 DATA FC,C6,3F,20,02,C6,15,10,EE,BC,01,39,00,00,98,00, 1110,55A
1120 DATA 0A,00,7F,00,01,04,00,00,01,00,00,00,00,00,00,00, 1120,0BF
1130 DATA 00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00, 1130,000
1140 DATA 00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00, 1140,000
1150 DATA 00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00, 1150,000
1160 DATA 8E,6E,00,CE,7E,00,EC,C1,ED,81,11,83,80,00,26,F6, 1160,793
1170 DATA CE,7E,00,EC,81,ED,C1,8C,71,00,26,F7,BD,7E,00,34, 1170,7FO
1180 DATA 04,8E,6E,00,CE,7E,00,EC,81,ED,C1,11,83,80,00,26, 1180,6A1
1190 DATA F6,35,04,7E,92,AB,00,00,00,00,00,00,00,00,00,00, 1190,2EA

```

## リスト3 LOADERアセンブル・リスト

```

0001 0000 ; RAM-data write on Floppy
0002 0000 ; Ver1.1 [date 83/01/07]
0003 0000 ; Debug [date 83/01/30]
0004 0000 ;
0005 7E00 ORG $7E00
0006 7E00 347F PSHS U,Y,X,DP,B,A,CC
0007 7E02 308D00E1 LEAX RCB,PC ; Seek Track-0
0008 7E06 ADFFFBFA JSR [$FBFA]
0009 7E0A 102500D3 LBSC ERR
000A 7E0E 860A LDA #$0A ; FAT Read
000B 7E10 A7B4 STA ,X
000C 7E12 ADFFFBFA JSR [$FBFA]
000D 7E16 102500C7 LBSC ERR
000E 7E1A 10AE02 LDY 2,X ; Free Cluster Search
000F 7E1D 3125 LEAY 5,Y
0010 7E1F 3008 LEAX 8,X ; [cluster buffer address]
0011 7E21 1F13 TFR X,U
0012 7E23 C6FF LDB #$FF
0013 7E25 4F LOOP1 : CLRA
0014 7E26 5C INCB
0015 7E27 C198 CMPB #$98
0016 7E29 102400B4 LBCC ERR
0017 7E2D A6AB LDA D,Y
0018 7E2F 81FF CMPA #$FF
0019 7E31 26F2 BNE LOOP1
001A 7E33 4F CLRA
001B 7E34 E7C0 STB ,U+
001C 7E36 E7B4 STB ,X
001D 7E38 30AB LEAX D,Y
001E 7E3A 6A8D00AB DEC COUNTER,PC
001F 7E3E 26E5 BNE LOOP1
0020 7E40 C6C7 LDB #$C7
0021 7E42 E7B4 STB ,X
0022 7E44 308D009F LEAX RCB,PC ; FAT Write
0023 7E48 6A84 DEC ,X
0024 7E4A ADFFFBFA JSR [$FBFA]
0025 7E4E 102500BF LBSC ERR
0026 7E52 8604 LDA #$04 ; Directory Read
0027 7E54 6C84 INC ,X
0028 7E56 A705 LOOP3 : STA 5,X
0029 7E58 ADFFFBFA JSR [$FBFA]
002A 7E5C 102500B1 LBSC ERR
002B 7E60 8680 LDA #$80
002C 7E62 EE02 LDU 2,X
002D 7E64 33C900B0 LEAU $80,U
002E 7E68 E6C6 LOOP2 : LDB A,U
002F 7E6A 2716 BEQ NEXT
0030 7E6C C1FF CMPB #$FF
0031 7E6E 2712 BEQ NEXT
0032 7E70 8B20 ADDA #$20
0033 7E72 28F4 BVC LOOP2
0034 7E74 A605 LDA 5,X
0035 7E76 4C INCA
0036 7E77 8111 CMPA #$11
0037 7E79 25D8 BCS LOOP3
0038 7E7B CC0101 LDD #$0101
0039 7E7E E706 STB 6,X
003A 7E80 20D4 BRA LOOP3
003B 7E82 33C6 NEXT : LEAU A,U ; Directory Set
003C 7E84 CCFE20 LDD #$FE20
003D 7E87 EDC1 STD ,U++
003E 7E89 8620 LDA #$20
003F 7E8B EDC1 STD ,U++
0040 7E8D EDC1 STD ,U++
0041 7E8F EDC1 STD ,U++
0042 7E91 8618 LDA #$18
0043 7E93 6FC0 LOOP4 : CLR ,U+
0044 7E95 4A DECA
0045 7E96 26FB BNE LOOP4
0046 7E98 A608 LDA 8,X ; [first cluster set]
0047 7E9A A7CBEE STA -18,U
0048 7E9D 6A84 DEC ,X ; Directory Write
0049 7E9F ADFFFBFA JSR [$FBFA]
004A 7EA3 253C BCS ERR
004B 7EA5 10BE0000 LDY #$0000 ; RAM-data Store
004C 7EA9 3308 LEAU 8,X
004D 7EAB A6C0 LOOP6 : LDA ,U+
004E 7EAD 1FB9 TFR A,B

```

## リスト3 LOADERアセンブル・リスト

```

004F 7EAF C401      ANDB ##01
0050 7EB1 58       LSLB
0051 7EB2 58       LSLB
0052 7EB3 58       LSLB
0053 7EB4 5C       INCB
0054 7EB5 E705     STB 5,X           ; Sector
0055 7EB7 44       LSRA
0056 7EB8 1F89     TFR A,B
0057 7EBA C401     ANDB ##01
0058 7EBC E706     STB 6,X           ; Side
0059 7EBE 44       LSRA
005A 7EBF 8B02     ADDA ##02
005B 7EC1 A704     STA 4,X           ; Track
005C 7EC3 C608     LDB ##08
005D 7EC5 10AF02   LOOP5:STY 2,X           ; store
005E 7EC8 ADFFFBFA JSR [F8FA]
005F 7ECC 2513     BCS ERR
0060 7ECE 31A90100 LEAY $100,Y
0061 7ED2 108C8000 CMPY ##8000
0062 7ED6 2407     BCC END
0063 7ED8 6C05     INC 5,X
0064 7EDA 5A       DECB
0065 7EDB 26E8     BNE LOOP5
0066 7EDD 20CC     BRA LOOP6
0067 7EDF 35FF     END :PULS CC,A,B,DP,X,Y,U,PC ; return
0068 7EE1 C615     ERR :LDB #21
0069 7EE3 7E92AB   JMP $92AB
006A 7EE6 10       CNTER:FCB $10
006B 7EE7 0800     RCB :FDB $0800
006C 7EE9 7F00     FDB $7F00
006D 7EEB 0101     FDB $0101
006E 7EED 0000     FDB $0000
006F 7EEF         CLBUF :RMB 16

```

## リスト4 SAVERアセンブル・リスト

```

0001 0000          ; RAM-data read on Flopy
0002 0000          ; Ver1.2 [date 83/01/12]
0003 0000          ; Debug [date 83/01/30]
0004 0000          ;
0005 0000          ; Disk Code Shelter Program
0006 7100          ORG $7100
0007 7100 8E6E00   LDX ##$6E00           ; Disk Code Shelter
0008 7103 CE7E00   LDU ##$7E00
0009 7106 ECC1     LOOP_1:LDD ,U++
000A 7108 EDB1     STD ,X++
000B 710A 11838000 CMPI ##$8000
000C 710E 26F6     BNE LOOP_1
000D 7110 CE7E00   LDU ##$7E00           ; Main Program Set
000E 7113 ECB1     LOOP_2:LDD ,X++
000F 7115 EDC1     STD ,U++
0010 7117 8C7100   CMPX ##$7100
0011 711A 26F7     BNE LOOP_2
0012 711C BD7E00   JSR $7E00
0013 711F 3404     FSHS B
0014 7121 8E6E00   LDX ##$6E00           ; Disk Code Return
0015 7124 CE7E00   LDU ##$7E00
0016 7127 ECB1     LOOP_3:LDD ,X++
0017 7129 EDC1     STD ,U++
0018 712B 11838000 CMPI ##$8000
0019 712F 26F6     BNE LOOP_3
001A 7131 3504     PULS B
001B 7133 7E92AB   JMP $92AB           ; Jump Error Subroutine
001C 7136          ;
001D 7136          ; Main Program
001E 7000          ORG $7000
001F 7000 10EF8D00B7 STS STACK,PC
0020 7005 10CEFC7F LDS ##$FC7F
0021 7009 308D00B3 LEAX RCB,PC
0022 700D 8604     LDA ##$04           ; Directory Read
0023 700F A705     LOOP3:STA 5,X
0024 7011 ADFFFBFA JSR [F8FA]
0025 7015 1025009C LBCL ERR
0026 7019 8680     LDA ##$80
0027 701B EE02     LDU 2,X
0028 701D 33C900B0 LEAU $80,U
0029 7021 E6C6     LOOP2:LDB A,U           ; File Search
002A 7023 C1FE     CMPB ##$FE
002B 7025 2606     BNE JUMP
002C 7027 6ABD009D DEC FILE,PC

```



## リスト4 SAVERアセンブル・リスト

```

002D 702B 2718      BEQ NEXT
002E 702D 6ABD00BD  JUMP :DEC DIRC,PC
002F 7031 277E      BEQ NoFILE
0030 7033 8B20      ADDA #$20
0031 7035 28EA      BVC LOOP2
0032 7037 A605      LDA 5,X
0033 7039 4C        INCA
0034 703A 8111      CMFA #$11
0035 703C 25D1      BCS LOOP3
0036 703E CC0101    LDD #$0101
0037 7041 E706      STB 6,X
0038 7043 20CA      BRA LOOP3
0039 7045 33C6      NEXT :LEAU A,U          ; First Cluster Search
003A 7047 E64E      LDB 14,U
003B 7049 318D007C  LEAY CLBUF,PC
003C 704D E7A0      STB ,Y+
003D 704F 860F      LDA #15
003E 7051 A7BC6B    STA <CINTER,PC
003F 7054 4F        CLRA
0040 7055 CE0100    LDU #$0100          ; FAT Read
0041 7058 EF05      STU 5,X
0042 705A ADFFFBFA  JSR [F$FBFA]
0043 705E 2555      BCS ERR
0044 7060 EE02      LDU 2,X          ; Target Clusters Search
0045 7062 3345      LEAU 5,U
0046 7064 E6CB      LOOP1 :LDB D,U
0047 7066 C198      CMPB #$98
0048 7068 244B      BCC ERR
0049 706A E7A0      STB ,Y+
004A 706C 6ABC50    DEC <CINTER,PC
004B 706F 26F3      BNE LOOP1
004C 7071 10BE0000  LDY #$0000          ; RAM-data Load
004D 7075 338D0050  LEAU CLBUF,PC
004E 7079 A6C0      LOOP6 :LDA ,U+
004F 707B 1FB9      TFR A,B
0050 707D C401      ANDB #$01
0051 707F 58        LSLB
0052 7080 58        LSLB
0053 7081 58        LSLB
0054 7082 5C        INCB
0055 7083 E705      STB 5,X          ; Sector
0056 7085 44        LSRA
0057 7086 1FB9      TFR A,B
0058 708B C401      ANDB #$01
0059 708A E706      STB 6,X          ; Side
005A 708C 44        LSRA
005B 708D 8B02      ADDA #$02
005C 708F A704      STA 4,X          ; Track
005D 7091 C608      LDB #$08
005E 7093 10AF02    LOOP5 :STY 2,X          ; load
005F 7096 ADFFFBFA  JSR [F$FBFA]
0060 709A 2519      BCS ERR
0061 709C 31A90100  LEAY $100,Y
0062 70A0 108C7E00  CMPY #$7E00
0063 70A4 2407      BCC END
0064 70A6 6C05      INC 5,X
0065 70A8 5A        DECB
0066 70A9 26EB      BNE LOOP5
0067 70AB 20CC      BRA LOOP6
0068 70AD 6EFFF8FC  END :JMP [F$FBFC]    ; BASIC hot start
0069 70B1 C63F      NoFILE:LDB #63
006A 70B3 2002      BRA RET
006B 70B5 C615      ERR :LDB #21
006C 70B7 10EE8C01  RET :LDS <STACK,PC
006D 70BB 39        RTS
006E 70BC 0000      STACK :FDB $0000
006F 70BE 98        DIRC :FCB $98
0070 70BF 00        CINTER :FCB $00
0071 70C0 0A00      RCB :FDB $0A00
0072 70C2 7F00      FDB $7F00
0073 70C4 0104      FDB $0104
0074 70C6 0000      FDB $0000
0075 70C8 01        FILE :FCB $01
0076 70C9          CLBUF :RMB 16

```

## 裏RAMの有効利用法

RAM  
FILE

ラム・ファイル

■渡辺満彦

このプログラムはFM-8の裏RAM31KバイトをRAM FILEとして活用するためのものです。裏RAMのREAD, WRITEにバブルファイル・コントロール・ルーチンを借用したので、バブルとほとんど同じ使い方ができます(詳しくは仕様を参照)。

ディスクのない方は高速ファイルを実感してみてください。ディスクのある方もカセット版でしか走らない大きなゲームも、ディスクから裏RAMに転送しておけばロードできるのです。

FM-8にも強力な後輩たちができました。FM-8もがんばらねば!

## 配線するところ

図1 回路図

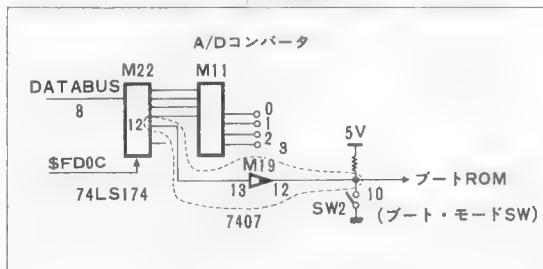
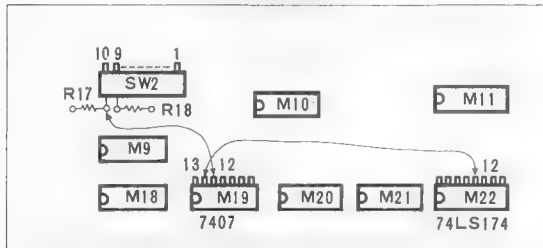


図2 実体図



回路図にあるように、A/Dコンバータでは使ってないビット6をオープン・コレクタのバッファを通してブート・モード・スイッチ10番に接続します。この配線によって\$FD0Cに\$40を書き込めば裏RAMに、\$00を書き込めばBASIC ROMにソフト切り替えができるようになります。

## 使い方(RAM)

まずブート・モードSW10のOFFを確かめます。リスト1を適当なアドレスにロードして、EXECロード先頭アドレス [RET] の入力で“Ready”が表示されればインターフェイスはOKです。なお、BUBINIO [RET] Y FILES “BUB0: [RET]”と入力して“1023Pages Free”と表示されれば、裏RAMに対する読み書きテストもOKになります。ここまですればバブルに関するすべてのコマンド、ステートメントが使えるようになります。後述の仕様と文法書などを参考に試してみてください(バブルは初めて使うときイニシャライズが必要です)。

## ROM版

実用性を考えてROM版も作ってみました。リスト2がそれです。アセンブラの都合でアドレスがずれていますがこれを\$F511~\$F5C7に書き込んで使います。この他にアドレス\$8A2B, Cの\$2725を\$2008に書き換える必要があります。結局2個のROMを焼き直すことになります。

このプログラムはユーザーエリアを使っていません。システム・エリアに数十バイトのテーブルが登録されるのでその分だけユーザーエリアが減少することになります。ROM版はインターフェース・プログラムのイニシャライズの必要はなく、電源ONですぐ使えます。

## リセット方法

ユーザープログラムが暴走して、リセット・スイッチを押してもBASICモードに戻れない場合、電源のOFF-ONまたはブート・モード・スイッチ10をONにしてリセットを押せばBASICモードに復帰できます。

なお、ブート・モード・スイッチ10をOFFにして電源を投入してもBASICは起動するはずですが、もし起動しない場合にはスイッチをONにして電源を投入するようにしてください。

## RAM FILE 仕様

- 電源OFFで消去されるが、リセットでは消去されない。
- 容量：992ページ
- 転送時間：約8ms/ページ
- デバイス名はBUB0、BUB1使用可。ただし、BUB1はBUB0内に二重定義したものであり、両デバイスの使用に当たっては注意が必要。これは2つのファイルを同時にOPEN可能とするため。
- 各バブルの割り当てアドレスとページ数、BUB0 \$8000-\$BFFF、992ページ、BUB1 \$C000-\$FBFF、480ページ。
- エラー表示、ページ・アドレスがBUB0は991、BUB1は479を越えた場合、Device I/O Errorが表示される。
- フロテクト、ブート・モード・スイッチ10をONとすれば、裏RAMの読み書きが禁止され、データが保護される。
- フリーページ数、FILESコマンドで表示されるフリーページ数が、31Kバイトのデバイスとなるために以下になる。  
BUB0：実際の値=表示値-32ページ  
BUB1：実際の値=表示値-544ページ
- ROM版おまけ、BUB1をBUB0上に二重定義しているが、\$FFFF番地を\$00以外の値とすることでユーザーエリア内にも定義することができる。\$FFFFの値がアドレス上位バイト、D位バイトは\$00になる。ただし、BASIC上限アドレスをCLEAR文でこのアドレス以下にしておくこと。

## コンパチビリティ

ハード的にはブート・モード・スイッチ10をONにしておけばBASICモードでのコンパチビリティは保たれます。ソフト的にはRAM版においては、インターフェイス・プログラムがユーザーエリアに置かれ、ROM版においてはシステム・エリアにテーブルが登録されるだけなので、一般的な使用法では問題は起きないと思います。

なお、この改造により、バブル・モードと予備モードがDIPスイッチによって選択することができなくなります。

## 終わりに

FM-8のファイル処理概要とこのプログラムについて簡単にふれておきます。

ファイル処理概要は図3のような構成となっています。ファイル処理のBASIC側側入力単位は、1ファイル名で処理されるプログラムまたはデータで、デバイス処理側はバイト単位となっています。この処理の内部でファイル・ディスクリプタ、パラメータの解析を行ない、ファイルのオープン、クローズ、1バイト・データの入力、出力ルーチンなどをCALLしています。

デバイス処理には各デバイスそれぞれに7つのエントリ・ポイントを持つエントリ・テーブルがあります。そのテーブルのポインタ・テーブルが\$06F9以降に置かれます。エントリ・テーブルは表1のようにになっています。

図 3 ファイル処理概要

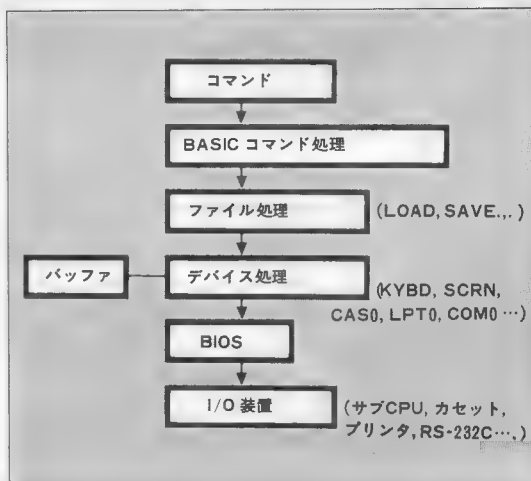


表 1 ファイル処理エントリ・テーブル

デバイス名	ファイル・オープン	ファイル・クローズ	1バイト入力	1バイト出力	?	?	?
CAS0	CD06	CDA6	CD19	CDF4	CE23	CCFA	CD02

1バイト入出力ルーチンはBASIC側からのデータを、バッファ上に各デバイスに決められたサイズにブロッキング、デブロッキングを行ないます。そのうえでBIOSをコールしてI/O装置に送受します。

このデバイス処理（私がかつてに命名）はCP/MのBDOSに相当するもので、これの使い方がわかるとCP/Mのシステム・コール（またはBDOSコール）と同じことができます。マシン語レベルから直接ファイルを扱えるようになります。このことはF-BASICのOSとしての機能を高めるものであり、実用的ソフトを作るには必要なものです。メーカーには是非このBDOSの詳しい使い方を一般に公開していただきたいものです。BASICも見方を変えて、ジョブ・コントロール・ランゲージとして見れば、CP/MのSUBMITなど問題にならないほど強力なものになるのですから。

最後にこのRAM FILEプログラムは、BIOSの中のバブル・ハンドラを裏RAMアクセス・プログラムに置き替えたものです。これが何かのお役に立てば幸いです。

## □参考資料

- 1) “FM-8活用研究”（回路図、BIOSリスト）
- 2) “FM-8のBIOS活用法”，インターフェース’82年2月号
- 3) FM-8 文法書、マニュアル
- 4) “MC6809/E マイクロプロセッサ プログラミングマニュアル”，CQ出版

## リスト1 RAM版

0001	0000	;	RAM FILE (RAM)	
0002	0000	;	83.1.20 BY M.WATANABE	
0003	0000	;		
0004	1800		ORG \$1800	
0005	1800 30BD001A	INIT	:LEAX BIQST,PC	;SET BIOS TRAP
0006	1804 9FDF		STX <\$DF	
0007	1806 8E0703		LDX #\$0703	;SET ENTORY TABLE
0008	1809 ECB1	STBL	:LDD ,X++	

```

0009 180B 26FC          BNE STBL
000A 180D 301E          LEAX -2,X
000B 180F 318D00BF      LEAY BUB0,PC
000C 1813 10AF81        STY ,X++
000D 1816 318D00DF      LEAY BUB1,PC
000E 181A 10AF84        STY ,X
000F 181D 39           RTS
0010 181E              ;
0011 181E              ; BIOS TRAP
0012 181E              ;
0013 181E 347F          BIOST :PSHS CC,A,B,DP,X,Y,U
0014 1820 E684          LDB ,X                ;GET REQ NO.
0015 1822 C104          CMPB #4                ;CLEAR BUBR ?
0016 1824 270D          BEQ CLR8
0017 1826 C107          CMPB #7                ;READ BUBR ?
0018 1828 271A          BEQ READB
0019 182A C106          CMPB #6                ;WRITE BUBR ?
001A 182C 2724          BEQ WRITB
001B 182E 357F          PULS CC,A,B,DP,X,Y,U
001C 1830 7EF2D8        JMP $F2D8            ;BIOS
001D 1833              ;
001E 1833              ; CLEAR BUBR
001F 1833              ;
0020 1833 C640          CLR8 :LDB #$40          ;RAM
0021 1835 F7FD0C        STB $FD0C
0022 1838 6F80          LOOPC :CLR ,X+
0023 183A 8CFC00        CMPX #$FC00
0024 183D 26F9          BNE LOOPC
0025 183F 7FFD0C        CLR $FD0C            ;ROM
0026 1842 35FF          PULS CC,A,B,DP,X,Y,U,PC
0027 1844              ;
0028 1844              ; READ WRITE BUBR
0029 1844              ;
002A 1844 C680          READB :LDB #$80          ;SET READ MODE
002B 1846 E78D004C      STB LOOP1+1,PC
002C 184A C6C0          LDB #$C0
002D 184C E78D0048      STB VSTA+1,PC
002E 1850 200C          BRA TRANSF
002F 1852              ;
0030 1852 C6C0          WRITB :LDB #$C0          ;SET WRITE MODE
0031 1854 E78D003E      STB LOOP1+1,PC
0032 1858 C680          LDB #$80
0033 185A E78D003A      STB VSTA+1,PC
0034 185E              ;
0035 185E EE02          TRANSF:LDD 2,X          ;BUFFER ADDRESS
0036 1860 EC06          LDD 6,X                ;COUNTS
0037 1862 8D4A          BSR ERCHK
0038 1864 EC04          LDD 4,X
0039 1866 C30001        ADDD #$0001
003A 1869 8D43          BSR ERCHK
003B 186B EC04          LDD 4,X
003C 186D E306          ADDD 6,X
003D 186F 8D3D          BSR ERCHK
003E 1871 EC04          LDD 4,X                ;PAGE ADDRESS
003F 1873 10AE06        LDY 6,X                ;PAGE COUNTS
0040 1876 3412          PSHS A,X
0041 1878 58           LSLB
0042 1879 49           ROLA
0043 187A 58           LSLB
0044 187B 49           ROLA
0045 187C 58           LSLB
0046 187D 49           ROLA
0047 187E 58           LSLB
0048 187F 49           ROLA
0049 1880 58           LSLB
004A 1881 49           ROLA
004B 1882 C38000        ADDD #$8000
004C 1885 68E0          LSL ,S+
004D 1887 2403          BCC NEXT
004E 1889 C34000        ADDD #$4000            ;UNIT1=$C000--$FBFF
004F 188C 1F01          NEXT :TFR D,X
0050 188E C640          LDB #$40            ;RAM
0051 1890 F7FD0C        STB $FD0C
0052 1893 C620          LOOP0 :LDB #32        ;1 PAGE
0053 1895 A680          LOOP1 :LDA ,X+        ;(,U+)
0054 1897 A7C0          VSTA :STA ,U+        ;(,X+)
0055 1899 5A           .DECB
0056 189A 26F9          BNE LOOP1
0057 189C 313F          LEAY -1,Y
0058 189E 26F3          BNE LOOP0
0059 18A0 3510          PULS X
005A 18A2 3501          PULS CC
005B 18A4 4F           CLRA
005C 18A5 A701          STA 1,X                ;CLEAR ERR
005D 18A7 B7FD0C        STA $FD0C            ;ROM
005E 18AA 1CFE          ANDC #$FE            ;CLEAR ERROR FLAG
005F 18AC 35FE          PULS A,B,DP,X,Y,U,PC

```

## リスト 1

```

0060 18AE          ;
0061 18AE          ; ERROR CHECK
0062 18AE          ;
0063 18AE 847F     ERCHK :ANDA #$7F
0064 18B0 1F02     TFR   D,Y
0065 18B2 A604     LDA   4,X          ;TEST DEV. NO.
0066 18B4 8580     BITA  #$80
0067 18B6 2707     BEQ   CBUB0
0068 18B8 108C01E0 CMPY  #$01E0          ;480
0069 18BC 2208     BHI   ERR
006A 18BE 39       RTS
006B 18BF 108C03E0 CBUB0 :CMPY #$03E0          ;992
006C 18C3 2201     BHI   ERR
006D 18C5 39       RTS
006E 18C6 3540     ERR   :PULS U
006F 18C8 3501     PULS CC
0070 18CA 8623     LDA   #35          ;PAGE ERROR CODE
0071 18CC A701     STA   1,X
0072 18CE 1A01     ORCC  #$01          ;SET ERROR FLAG
0073 18D0 35FE     PULS A,B,DP,X,Y,U,PC
0074 18D2          ;
0075 18D2          ; ENTORY TABLES
0076 18D2          ;
0077 18D2 42554230 BUB0 :FCC /BUB0/
0078 18D6 ECBE     FDB  $ECBE
0079 18D8 EDB0     FDB  $EDB0
007A 18DA EE34     FDB  $EE34
007B 18DC EEAC     FDB  $EEAC
007C 18DE EEFF     FDB  $EEFF
007D 18E0 EF0B     FDB  $EF0B
007E 18E2 EF37     FDB  $EF37
007F 18E4 00       FCB  $00
0080 18E5 FC00     FDB  $FC00
0081 18E7          RMB  18
0082 18F9 42554231 BUB1 :FCC /BUB1/
0083 18FD ECBE     FDB  $ECBE
0084 18FF EDB0     FDB  $EDB0
0085 1901 EE34     FDB  $EE34
0086 1903 EEAC     FDB  $EEAC
0087 1905 EEFF     FDB  $EEFF
0088 1907 EF0B     FDB  $EF0B
0089 1909 EF37     FDB  $EF37
008A 190B 00       FCB  $00
008B 190C FC20     FDB  $FC20
008C 190E          RMB  19

```

## RAM-FILEダンプ (参考)

```

10 DIM A%(15) :C=&H8000
20 FOR I=0 TO 99:1
30 BUBRO,I,A% :BUBRO,I,A%
40 AD=C+I*32 :PRINT HEX$(AD) " ";
50 FOR J=0 TO 15
60 PRINT RIGHT$("000"+HEX$(A%(J)),4) " ";
70 IF J=7 THEN PRINT :PRINT " ";
80 NEXT J
90 FOR J=1 TO 32
100 P%=MID$(A%,J,1) :D=ASC(P%)
110 IF D>32 AND D<127 THEN PRINT P% ; ELSE PRINT ". ";
120 NEXT J
130 PRINT
140 NEXT I

```

## リスト 2 ROM版 \$F511~\$F5C7に書き込む

```

0001 0000          ; RAM FILE (ROM)
0002 0000          ; '83.1.22 BY M.WATANABE
0003 0000          ;
0004 1811          ORG  $1811
0005 1811 318C22   LEAY  <CLEARP,PC          ;LOAD CLEAR PROG.
0006 1814 8608     LDA   #8
0007 1816 EEA3     LOOPTC:LDU  ,--Y
0008 1818 3440     PSHS  U
0009 181A 4A       DECA
000A 181B 26F9     BNE  LOOPTC
000B 181D BEB000   LDX   #$B000
000C 1820 ADE4     JSR   ,S
000D 1822 32EB10   LEAS  16,S          ;CLEAR BUBR
000E 1825 39       RTS                ;RESTORE STACK
000F 1826          ;
0010 1826 C640     LDB   #$40          ;RAM
0011 1828 D70C     STB  <$0C
0012 182A 6F80     LOOPTC:CLR  ,X+
0013 182C 8CFC00   CMPX  #$FC00
0014 182F 26F9     BNE  LOOPTC

```

```

0015 1831 4F          CLRA
0016 1832 970C        STA  <$0C          ;ROM
0017 1834 39          RTS
0018 1835 00          FCB  $00
0019 1836             CLEARP:RMB 1
001A 1837             ;
001B 1837             ; ADDRESS CHECK
001C 1837             ;
001D 1836             ORG  $1836
001E 1836 847F        ADCHK :ANDA #$7F
001F 1838 1F02        TFR  D,Y
0020 183A A604        LDA  4,X          ;TEST DEV. NO.
0021 183C 8580        BITA  #$80
0022 183E 2707        BEQ  CBUB0
0023 1840 108C01E0     CMPLY #$01E0     ;480
0024 1844 2208        BHI  ERR
0025 1846 39          RTS
0026 1847 108C03E0     CBUB0 :CMPLY #$03E0 ;992
0027 184B 2201        BHI  ERR
0028 184D 39          RTS
0029 184E 32EB18       ERR  :LEAS 24,S
002A 1851 8623        LDA  #35
002B 1853 39          RTS
002C 1854             ;
002D 1854             ; R/W BUBR
002E 1854             ;
002F 1857             ORG  $1857
0030 1857 318C6D       LEAY <TRANP,PC   ;LOAD TRANS. PROG.
0031 185A 860B        LDA  #11
0032 185C EEA3        LOOP1 :LDU ,--Y
0033 185E 3440        PSHS U
0034 1860 4A          DECA
0035 1861 26F9        BNE  LOOP1
0036 1863 E684        LDB  ,X          ;TEST R/W
0037 1865 C107        CMPB #7
0038 1867 2708        BEQ  LDR          ;READ
0039 1869 C6C0        LDB  #$C0        ;WRITE
003A 186B E767        STB  7,S
003B 186D C680        LDB  #$80
003C 186F E769        STB  7,S
003D 1871 EC04        LDR  :LDD 4,X     ;TEST ADDRESS
003E 1873 C30001      ADDD #$0001
003F 1876 BDBE        BSR  ADCHK
0040 1878 EC06        LDD  6,X
0041 187A BDBA        BSR  ADCHK
0042 187C EC04        LDD  4,X
0043 187E E306        ADDD 6,X
0044 1880 BDB4        BSR  ADCHK
0045 1882 EE02        LDU  2,X          ;BUFFER ADDRESS
0046 1884 EC04        LDD  4,X          ;PAGE ADDRESS
0047 1886 10AE06      LDY  6,X          ;PAGE COUNTS
0048 1889 3402        PSHS A
0049 188B 58          LSLB             ;D*32
004A 188C 49          ROLA
004B 188D 58          LSLB
004C 188E 49          ROLA
004D 188F 58          LSLB
004E 1890 49          ROLA
004F 1891 58          LSLB
0050 1892 49          ROLA
0051 1893 58          LSLB
0052 1894 49          ROLA
0053 1895 68E0        LSL  ,S+
0054 1897 240E        BCC  B0
0055 1899 7DFFF0      TST  $FFFF0
0056 189C 2705        BEQ  DFRT
0057 189E BBFFF0      ADDA $FFFF0
0058 18A1 2006        BRA  EXEC
0059 18A3 8BC0        DFRT :ADDA #$C0
005A 18A5 2002        BRA  EXEC
005B 18A7 8B80        B0  :ADDA #$80
005C 18A9 1F01        EXEC :TFR D,X
005D 18AB ADE4        JSR  ,S          ;STACK PROG.
005E 18AD 32EB16      LEAS 22,S        ;RESTORE STACK
005F 18B0 39          RTS
0060 18B1             ;
0061 18B1             ; STACK PROGRAM
0062 18B1             ;
0063 18B1 C640        LDB  #$40          ;RAM
0064 18B3 D70C        STB  <$0C
0065 18B5 C620        LOOP0:LDB #32     ;1 PAGE
0066 18B7 A680        LOOPR1:LDA ,X+    ;(,U+)
0067 18B9 A7C0        STA  ,U+          ;(,X+)
0068 18BB 5A          DECB
0069 18BC 26F9        BNE  LOOPR1
006A 18BE 313F        LEAY -1,Y
006B 18C0 26F3        BNE  LOOP0

```

## リスト 2

```

006C 18C2 4F          CLRA
006D 18C3 970C        STA  <$0C          ;ROM
006E 18C5 39          RTS
006F 18C6 00          FCB  $00
0070 18C7          TRANP :RMB 1

```

## RAM-FILE ダンプ例

```

RUN
B000 564F 4C30 3030 3030 03E0 000E 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000 VOL000000.....
B020 4455 4D50 4220 2020 0041 000D 1800 0000
      0000 0000 0000 0000 0000 0000 0000 DUMPB....A.....
B040 0D31 3020 2044 494D 2041 252B 3135 2920
      3A43 3D26 4838 3030 300D 3230 2020 464F .10..DIM.A%(15).:C=&H8000.20..FO
B060 5220 493D 3020 544F 2039 3931 0D33 3020
      2020 4255 4252 302C 492C 4125 203A 4255 R.I=0.TD.991.30...BUBRO,I,A%.:BU
B080 4252 302C 492C 4124 0D34 3020 2020 4144
      3D43 2B49 2A33 3220 3A50 5249 4E54 2048 BR0,I,A$.40...AD=C+I*32.:PRINT.H
B0A0 455B 2428 4144 2922 2020 223B 0D35 3020
      2020 464F 5220 4A3D 3020 544F 2031 350D EX$(AD) "...:50...FOR.J=0.TD.15.
B0C0 3630 2020 2020 5052 494E 5420 5249 4748
      5424 2822 3030 3022 2B48 455B 2428 4125 60...PRINT.RIGHT$("000"+HEX$(A%
B0E0 284A 2929 2C34 2922 2022 3B0D 3730 2020
      2020 4946 204A 3D37 2054 4845 4E20 5052 (J)),4)".":70...IF.J=7.THEN.FR
B100 494E 5420 3A50 5249 4E54 2022 2020 2020
      2020 223B 0D38 3020 2020 4E45 5854 204A INT.:PRINT.".....":80...NEXT.J

```

## R A N D O M B O X

2<sup>n</sup>-1計算プログラム

■関 牧人

I/O'83年3月号に最大素数のことが、載っていましたが、28番めのメルセンヌ素数である、2<sup>86243</sup>-1(25962桁)が'82年の暮に発見されたというのを知ったので、その値を計算できるプログラムを作りました。

このプログラムは2<sup>n</sup>-1をFM-8で計算するもので、RUNさせてnを入力すると、計算結果と桁数と計算時間をプリンタに出力します。86432を入力すると7時間53分後に出力します(3月号の2<sup>44497</sup>-1は2時間6分)。

ワーク・エリアは&H1000からnに比例して広がり、そのバイト数は実行後&HFFA,Bに入っています。

## サンプル印字

```

2^ 125 -1 ..... 38 77 00:00:00
4253529586511730793292182592971026431

```

## 最大素数を求めるプログラム

```

100 '-----$$$$ 2^N-1 $$$-----
110 FOR I=&HFF90 TO &HFFF:READ A$:POKE I,VAL("&H"+A$):NEXT I
120 INPUT N:IF N<1 THEN 120
130 POKE &HFFA,FIX(N/256/2):M=FIX(N-FIX(N/256/2)*256/2)
140 POKE &HFF5,FIX(N/256):POKE &HFF6,M-FIX(N/256)*256
150 TIME$="00:00:00":EXEC&HFF90:T$=TIME$
160 B=PEEK(&HFFA)*256+PEEK(&HFFB):E=B-1:&H1000
170 D$=STR$(PEEK(E)):C=LEN(D$)-1
180 OPEN "O":L"LP10":
190 PRINT# 1," 2^N-1 .....":C+2*(B-1):"77 "T$
200 PRINT# 1:PRINT# 1,RIGHT$(" ")+RIGHT$(D$,C/2):E=E-1
210 WHILE E=>&H1000
220 D$=STR$(PEEK(E)):C=LEN(D$)-1
230 PRINT# 1,RIGHT$("0")+RIGHT$(D$,LEN(D$)-1/2):
240 E=E-1:WEND
250 PRINT# 1:PRINT# 1:PRINT# 1:CLOSE 1:GO TO 120
260 DATA BE,00,01,BF,0F,FA,B8,01,B7,10,00,10,BE,10,00,7F
270 DATA 0F,F7,FD,0F,FA,B8,0F,FB,A5,A4,AB,A4,BB,0F,F7,7F
280 DATA 0F,F7,B1,26,05,05,7C,0F,F7,B0,6A,A7,A0,CC,FF,FF
290 DATA F3,0F,F8,26,E0,C6,01,F1,0F,F7,26,09,BE,0F,FA,3A
300 DATA BF,0F,FA,E7,A4,CC,FF,FF,F3,0F,F5,FD,0F,F5,26,BB
310 DATA B6,0F,F4,27,0B,7A,0F,F4,BE,FF,FF,BF,0F,F5,20,AB
320 DATA 7A,10,00,39,00,00,00,00,00,00,00,00,00,00,00,00

```



FM-8 漢字印刷

MP-80TYPEII (MB27401) 漢字印刷

漢字印刷

FM-8の専用プリンタで漢字を印刷するのに、ハードコピーを取ってはいは、

- ①非常に遅い。
- ②用紙いっぱいに使えない。
- ③コピーとコピーの間に間隔があく。

という欠点があり、思うようにいきません。そこで、漢字を印刷するプログラムを作ってみました。

## 特徴

本プログラムはBASICからマシン語を呼んで印刷するので、次のような特徴を持っています。

- ①ポジション・インディペンデントである。
- ②漢字の角度を4方向に指定でき、縦書きができる。
- ③1行分の漢字コードを配列に納め、マシン語に引き渡す。
- ④サブルーチンであり、ワード・プロセッサへの移植および漢字処理プログラムの開発などが容易にできる。

## 使い方

まず、CLEAR、&H7CFFを実行し、マシン語を入力してテープその他のデバイスにファイル名「KPRINT」でSAVEしておきます。BASICの方は、サンプル・プログラムのようにイニシャライズしておきます（配列名は「K」でなくてもよい）。マシン語が人力済みの場合はLOADM 命令は不要です。

変数K(0)には角度コード0～3と、漢字16ドットに対する漢字間の縦横間隔0～15ドットを、

$$K(0) = (\text{角度コード}) \times 256 + (\text{縦間隔}) \times 16 + (\text{横間隔})$$

のように納めます。

角度コードはSYMBOL文と同じです。そしてK(1)～K(60)に1行分の漢字コードを代入し、K=FNK(0)でプリントに打ち出されます（ここでのKはダミーであり、何でもよい）。

なお、1行の文字数は横の間隔によって表1に示すとおりです。間隔は通常縦・横ともに4ドットくらいがよいでしょう。ライン印刷済みの用紙を使う場合、縦間隔は8ドットにしてください。

マシン語実行後、1行の紙送り量は（縦間隔+8）ドットにセットされているので、漢字印刷のあとに普通文字を印刷する場合、PRINT #1, CHR\$(27)+"2";を実行して紙送り量を12ドットに戻しておく必要があります。

なお、マシン語実行によって配列の内容は変化しません。またDISK BASICの場合、マシン語は&H6D00から入力してください。

このプログラムはビット・イメージを使って無理に漢字をプリントしているため、字が大きい、遅いという欠点がありますが、これを解決するには漢字プリンタを利用する以外、やむを得ないでしょう。

## プログラムの説明

倍密ビット・イメージは横960ドットあるので、1行には $960 \div (16 + \text{横間隔})$ 文字の漢字が印刷可能になります。漢字は上下8ドットずつ2度に分けて印刷し、縦間隔分紙送りして間隔をあけます。

BIOSを使って取り出した漢字のパターン・データは図3のように横8ドット分が左右交互になっています。この

表 1

間隔	文字数
0	60
1	56
2	53
3	50
4	48
5	45
6	43
7	41
8	40
9	38
10	36
11	35
12	34
13	33
14	32
15	30

図 1 K(0) の内容



図 2 K=USR(VARPTR[K(0)])実行時

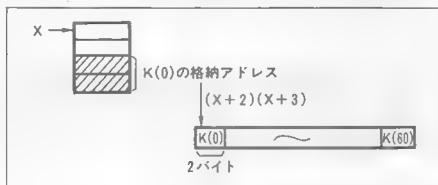


図 3 BIOSからの漢字データ

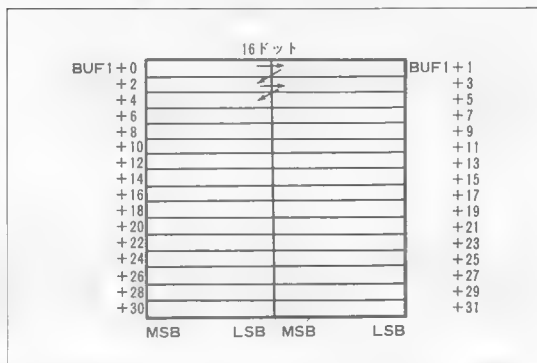
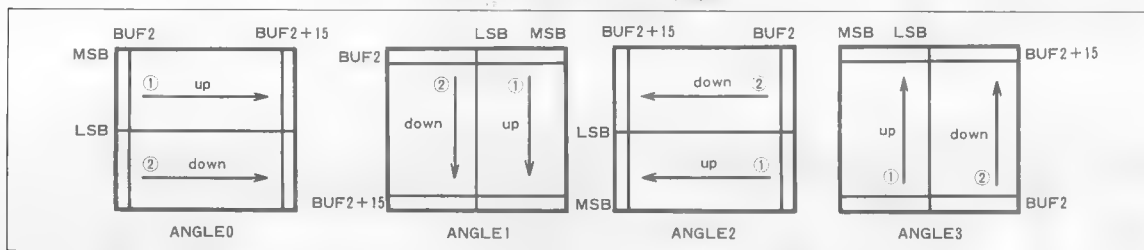


図 4 プリント出力用に変換



データを角度コード別に図4のようにビット・イメージ用にデータ変換し、プリンタに送ります。

データをマシン語に引き渡すのに、あるアドレスにPOKEしていく方法では手間と時間がかかるため、配列に納める方法が最も能率的です。USR命令を実行するとXレジスタが変数の格納アドレスを示しますが、配列の場合、先頭アドレスを示しません。そこでVARPTR命令で取り出した配列の先頭アドレスをUSR命令でマシン語に引き渡すのです。そうするとXレジスタが示すアドレス2バイトに配列の先頭アドレスを納めることができます(図2)。

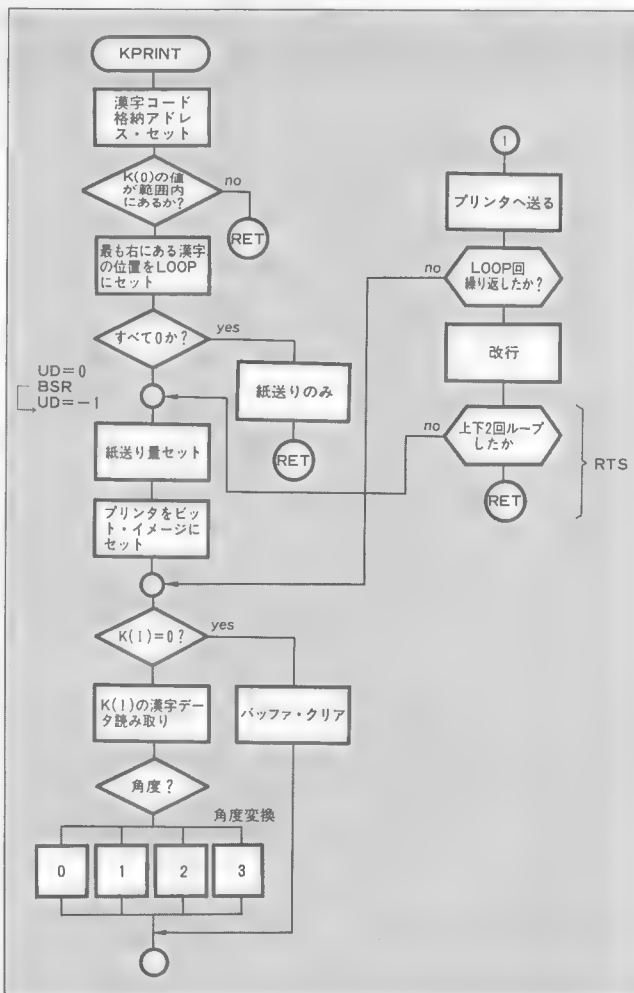
## 終わりに

私は大阪FM-8ユーザーズクラブを通じて、Z80のマシン語は知っていても6809は知らないという人が意外と多いのには驚きました。6809は命令が整っていて使い易く、理解できれば素晴らしいCPUであることがわかります。6809の命令は60個のニモニックで覚えられる範囲にあり、しかも組み合わせは1,500命令もあります。Z80のように存在するつもりで使っていた命令がアセンブルしてみても存在しないことが初めてわかった、などという事はないのです。FM-8ユーザーの皆さん、6809をマスターしよう! 決して難しくはないのです。

### 参考文献

- 1) "F-BASIC文法書": 富士通
- 2) "BIOSの使い方": インターフェース, '82年2月号

図 5 フローチャート



## サンプル・プログラム

```

10 ? DAMO 1
20 CLEAR,&H7CFF:LOADM"KPRINT"
30 DEFINTA-Z:DIMK(60)
40 DEFUSR=&H7D00:DEFFNK(K)=USR(VARPTR(K(0)))
50 FORI=1TO48:READA$:K(I)=&H2400+VAL("&H"+A$):NEXTI
60 FORI=0TO3:K(0)=I*256+4*16+4:K=FNK(0):NEXTI
70 DATA 22,24,26,28,2A,0,2B,2D,2F,31,33,0,35,37,39,3B,3D,0,3F,41,44,46,48,0,4A,4
B,4C,4D,4E,0,4F,52,55,58,5B,0,5E,5F,60,61,62,0,64,66,68,0,0,0

```

## サンプル印字例

```

あいうえお かきくけこ さしすせそ たちつてと なにぬねの はひふへほ まみむめも やゆよ
あいうえお かきくけこ さしすせそ たちつてと なにぬねの はひふへほ まみむめも やゆよ
あいうえお かきくけこ さしすせそ たちつてと なにぬねの はひふへほ まみむめも やゆよ
あいうえお かきくけこ さしすせそ たちつてと なにぬねの はひふへほ まみむめも やゆよ

```

## 漢字印刷アセンブル・リスト

```

0001 0000 ;
0002 0000 ; KPRINT
0003 0000 ;
0004 7D00 DRG $7D00
0005 7D00 AE02 LDX 2,X ; (CODEAD)<---ADRS K(1)
0006 7D02 3002 LEAX 2,X
0007 7D04 AF8D025B STX CODEAD,PC
0008 7D08 EC1E LDD -2,X ; IF K(0)>$03FF THEN RETURN
0009 7D0A 10830400 CMPD ##0400
000A 7D0E 2501 BCS JAMP1
000B 7D10 39 RTS
000C 7D11 A78D0250 JAMP1 : STA ANGLE,PC ; ANGLE,HSPACE,VSPACE SET
000D 7D15 1F98 TFR B,A
000E 7D17 C40F ANDB ##0F
000F 7D19 E78D0249 STB HSPACE,PC
0010 7D1D 44 LSRA
0011 7D1E 44 LSRA
0012 7D1F 44 LSRA
0013 7D20 44 LSRA
0014 7D21 A78D0242 STA VSPACE,PC
0015 7D25 318D0291 LEAY MOZISU,PC ; (LOOP)<---MOST RIGHT POSITION
0016 7D29 E6A5 LDB B,Y
0017 7D2B 301E LEAX -2,X
0018 7D2D 58 ASLB
0019 7D2E 10AE85 LOOP1 : LDY B,X
001A 7D31 2607 BNE JAMP2
001B 7D33 5A DECB
001C 7D34 5A DECB
001D 7D35 26F7 BNE LOOP1
001E 7D37 1600F5 LBRA NULL ; IF ALLCLEAR GOTO NULL
001F 7D3A 54 JAMP2 : LSRB
0020 7D3B E78D0229 STB LOOP,PC
0021 7D3F 6F8D0226 CLR UD,PC ; FIRST LOOP SET
0022 7D43 8D04 BSR JAMP3 ; ***** 2LOOP
0023 7D45 638D0220 COM UD,PC ; SECOND LOOP SET
0024 7D49 308D0228 JAMP3 : LEAX RCB,PC ; LF SIZE,LPT SET
0025 7D4D 860E LDA #PCMD
0026 7D4F A784 STA ,X
0027 7D51 318D021C LEAY CTRL,PC
0028 7D55 10AF02 STY 2,X
0029 7D58 CC1B41 LDD ##1B41
002A 7D5B EDA4 STD ,Y
002B 7D5D 8608 LDA #8
002C 7D5F 6D8D0206 TST UD,PC
002D 7D63 2704 BEQ JAMP4
002E 7D65 AB8D01FE ADDA VSPACE,PC
002F 7D69 A722 JAMP4 : STA 2,Y
0030 7D6B CC0003 LDD #3
0031 7D6E ED04 STD 4,X
0032 7D70 BDF2DB JSR BIOS
0033 7D73 308D01FE LEAX RCB,PC ; bit image LPT SET
0034 7D77 860E LDA #PCMD
0035 7D79 A784 STA ,X

```

```

0036 7D7B 318D01F2      LEAY CTRL,PC
0037 7D7F 10AF02        STY 2,X
0038 7D82 CC0004        LDD #4
0039 7D85 ED04          STD 4,X
003A 7D87 CC1B4C        LDD #1B4C
003B 7D8A EDA4          STD ,Y
003C 7D8C A68D01DB      LDA LOOP,PC
003D 7D90 C610          LDB #16
003E 7D92 EB8D01D0      ADDB HSPACE,PC
003F 7D96 3D            MUL
0040 7D97 1E89          EXG A,B
0041 7D99 ED22          STD 2,Y
0042 7D9B BDF2DB        JSR BIOS
0043 7D9E 6F8D01C8      CLR COUNT,PC      ;COUNTER CLEAR
0044 7DA2 AE8D01BD      NEXT :LDX CODEAD,PC ;K(A) CODE READ
0045 7DA6 A68D01C0      LDA COUNT,PC
0046 7DAA 4B            ASLA
0047 7DAB 10AE86        LDY A,X
0048 7DAE 272F          BEQ CLEAR      ;IF CODE=0 THEN BUF CLR
0049 7DB0 308D01C1      LEAX RCB,PC    ;K PATTERN DATA INPUT
004A 7DB4 8616          LDA #KCMD
004B 7DB6 A784          STA ,X
004C 7DB8 10AF04        STY 4,X
004D 7DBB 318D01BC      LEAY BUF1,PC
004E 7DBF 10AF02        STY 2,X
004F 7DC2 BDF2DB        JSR BIOS
0050 7DC5 308C22        LEAX <RETURN,PC    ;RETURN ADRS PUSH
0051 7DC8 3410          PSHS X
0052 7DCA A68D0197      LDA ANGLE,PC    ;ANGLE CHANGE SUB CALL
0053 7DCE 10270084      LBEQ ANGLE0
0054 7DD2 4A            DECA
0055 7DD3 102700EF      LBEQ ANGLE1
0056 7DD7 4A            DECA
0057 7DD8 1027010D      LBEQ ANGLE2
0058 7DDC 16016A        LBRA ANGLE3
0059 7DDF 308D01B8      CLEAR :LEAX BUF2,PC ;BUF CLEAR
005A 7DE3 8610          LDA #16
005B 7DE5 6F80          LOOP2 :CLR ,X+
005C 7DE7 4A            DECA
005D 7DEB 26FB          BNE LOOP2
005E 7DEA 308D0187      RETURN:LEAX RCB,PC ;CHANGE DATA,HSPACE LPT OUT
005F 7DEE 860E          LDA #PCMD
0060 7DF0 A784          STA ,X
0061 7DF2 318D01A5      LEAY BUF2,PC
0062 7DF6 10AF02        STY 2,X
0063 7DF9 C610          LDB #16
0064 7DFB EB8D0167      ADDB HSPACE,PC
0065 7DFE 4F            CLRA
0066 7E00 ED04          STD 4,X
0067 7E02 BDF2DB        JSR BIOS
0068 7E05 6C8D0161      INC COUNT,PC    ;COUNTER INCREMENT
0069 7E09 A68D015D      LDA COUNT,PC    ;IF COUNT=LOOP THEN NEXT
006A 7E0D A18D0157      CMPA LOOP,PC
006B 7E11 26BF          BNE NEXT
006C 7E13 318D015A      LEAY CTRL,PC    ;LPT LF
006D 7E17 860A          LDA #10A
006E 7E19 A7A4          STA ,Y
006F 7E1B 308D0156      LEAX RCB,PC
0070 7E1F 860E          LDA #PCMD
0071 7E21 A784          STA ,X
0072 7E23 10AF02        STY 2,X
0073 7E26 CC0001        LDD #1
0074 7E29 ED04          STD 4,X
0075 7E2B BDF2DB        JSR BIOS
0076 7E2E 39            RTS      ;SECOND LOOP OR BASIC
0077 7E2F              ;
0078 7E2F              ;SUB ROUTINE
0079 7E2F              ;
007A 7E2F 318D013E      NULL :LEAY CTRL,PC ;LPT LF (16+VSPACE dot)
007B 7E33 CC1B41        LDD #1B41
007C 7E36 EDA4          STD ,Y
007D 7E38 A68D012B      LDA VSPACE,PC
007E 7E3C 8B10          ADDA #16
007F 7E3E C60A          LDB #10A
0080 7E40 ED22          STD 2,Y
0081 7E42 308D012F      LEAX RCB,PC
0082 7E46 860E          LDA #PCMD
0083 7E4B A784          STA ,X

```

```

0084 7E4A 10AF02      STY 2,X
0085 7E4D CC0004      LDD #4
0086 7E50 ED04        STD 4,X
0087 7E52 BDF2D8      JSR BIOS
0088 7E55 39          RTS
                                ;BASIC RETURN
                                ;ANGLE 0 CHANGE (BUF2<---BUF1)
0089 7E56 308D0141    ANGLE0:LEAX BUF2,PC
008A 7E5A AF8D010F      STX TO,PC
008B 7E5E 308D0119      LEAX BUF1,PC
008C 7E62 6D8D0103      TST UD,PC
008D 7E66 2703          BEQ JAMP5
008E 7E68 308B10          LEAX 16,X
008F 7E6B AF8D00FC      JAMP5 :STX FROM,PC
0090 7E6F 8D14          BSR CHANGO
0091 7E71 AE8D00FB      LDX TO,PC
0092 7E75 3008          LEAX 8,X
0093 7E77 AF8D00F2      STX TO,PC
0094 7E7B AE8D00EC      LDX FROM,PC
0095 7E7F 3001          LEAX 1,X
0096 7E81 AF8D00E6      STX FROM,PC
0097 7E85 6F8D00E7      CHANGO:CLR YCT,PC
0098 7E89 6F8D00E2      LOOP3 :CLR XCT,PC
0099 7E8D AE8D00DA      LOOP4 :LDX FROM,PC
009A 7E91 A68D00DA      LDA XCT,PC
009B 7E95 4B          ASLA
009C 7E96 A686          LDA A,X
009D 7E98 E68D00D4      LDB YCT,PC
009E 7E9C 2704          BEQ JAMP6
009F 7E9E 4B          LOOP5 :ASLA
00A0 7E9F 5A          DECB
00A1 7EA0 26FC          BNE LOOP5
00A2 7EA2 4B          JAMP6 :ASLA
00A3 7EA3 AE8D00C6      LDX TO,PC
00A4 7EA7 A68D00C5      LDA YCT,PC
00A5 7EAB 6986          ROL A,X
00A6 7EAD 6C8D00BE      INC XCT,PC
00A7 7EB1 A68D00BA      LDA XCT,PC
00A8 7EB5 8108          CMPA #8
00A9 7EB7 26D4          BNE LOOP4
00AA 7EB9 6C8D00B3      INC YCT,PC
00AB 7EBD A68D00AF      LDA YCT,PC
00AC 7EC1 8108          CMPA #8
00AD 7EC3 26C4          BNE LOOP3
00AE 7EC5 39          RTS
                                ;ANGLE 1 CHANGE
00AF 7EC6 318D00D1    ANGLE1:LEAY BUF2,PC
00B0 7ECA 308D00AD      LEAX BUF1,PC
00B1 7ECE 6D8D0097      TST UD,PC
00B2 7ED2 2602          BNE JAMP7
00B3 7ED4 3001          LEAX 1,X
00B4 7ED6 8610          JAMP7 :LDA #16
00B5 7ED8 C608          LOOP6 :LDB #8
00B6 7EDA 6884          LOOP7 :ASL ,X
00B7 7EDC 66A4          ROR ,Y
00B8 7EDE 5A          DECB
00B9 7EDF 26F9          BNE LOOP7
00BA 7EE1 3002          LEAX 2,X
00BB 7EE3 3121          LEAY 1,Y
00BC 7EE5 4A          DECA
00BD 7EE6 26F0          BNE LOOP6
00BE 7EE8 39          RTS
                                ;ANGLE 2 CHANGE
00BF 7EE9 308D00AE    ANGLE2:LEAX BUF2,PC
00C0 7EED AF8C7D      STX <TO,PC
00C1 7EF0 308D0087      LEAX BUF1,PC
00C2 7EF4 3001          LEAX 1,X
00C3 7EF6 6D8C70      TST <UD,PC
00C4 7EF9 2603          BNE JAMP8
00C5 7EFB 308B10          LEAX 16,X
00C6 7EFE AF8C6A      JAMP8 :STX <FROM,PC
00C7 7F01 8D10          BSR CHANG2
00C8 7F03 AE8C67      LDX <TO,PC
00C9 7F06 3008          LEAX 8,X
00CA 7F08 AF8C62      STX <TO,PC
00CB 7F0B AE8C5D      LDX <FROM,PC
00CC 7F0E 301F          LEAX -1,X
00CD 7F10 AF8C5B      STX <FROM,PC
00CE 7F13 6F8C5A      CHANG2:CLR <YCT,PC
00CF 7F16 6F8C56      LOOP8 :CLR <XCT,PC
00D0 7F19 AE8C4F      LOOP9 :LDX <FROM,PC
00D1 7F1C A68C50      LDA <XCT,PC

```

```

00D2 7F1F 48          ASLA
00D3 7F20 A686        LDA  A,X
00D4 7F22 E68C4B      LDB  <YCT,PC
00D5 7F25 2704        BEQ  JAMP9
00D6 7F27 44          LOOP10:LSRA
00D7 7F28 5A          DECB
00D8 7F29 26FC        BNE  LOOP10
00D9 7F2B 44          JAMP9 :LSRA
00DA 7F2C AE8C3E      LDX  <TD,PC
00DB 7F2F A68C3E      LDA  <YCT,PC
00DC 7F32 6686        ROR  A,X
00DD 7F34 6C8C38      INC  <XCT,PC
00DE 7F37 A68C35      LDA  <XCT,PC
00DF 7F3A 8108        CMPA #8
00E0 7F3C 26DB        BNE  LOOP9
00E1 7F3E 6C8C2F      INC  <YCT,PC
00E2 7F41 A68C2C      LDA  <YCT,PC
00E3 7F44 8108        CMPA #8
00E4 7F46 26CE        BNE  LOOP8
00E5 7F48 39          RTS
00E6 7F49 318C4F      ANGLE3:LEAY <BUF2,PC      ;ANGLE 3 CHANGE
00E7 7F4C 308C2C      LEAX <BUF1,PC
00E8 7F4F 308B20      LEAX 32,X
00E9 7F52 6D8C14      TST  <UD,PC
00EA 7F55 2702        BEQ  JAMP10
00EB 7F57 3001        LEAX 1,X
00EC 7F59 8610        JAMP10:LDA #16
00ED 7F5B E683        LOOP11:LDB ,--X
00EE 7F5D E7A0        STB  ,Y+
00EF 7F5F 4A          DECA
00F0 7F60 26F9        BNE  LOOP11
00F1 7F62 39          RTS
00F2 7F63            ;
00F3 7F63            ;WORK AREA
00F4 7F63            ;
00F5 7F63            BIOS :EQU $F2D8      ;BIOS START ADRS
00F6 7F63            PCMD :EQU 14        ;BIOS PRINTER COMMAND
00F7 7F63            KCMD :EQU 22        ;BIOS KANZI COMMAND
00F8 7F63 0000        CODEAD:FDB 0      ;K(1) ADRS
00F9 7F65 00          ANGLE :FCB 0      ;ANGLE CODE
00FA 7F66 00          HSPACE:FCB 0      ;HSPACE SIZE
00FB 7F67 00          VSPACE:FCB 0      ;VSPACE SIZE
00FC 7F68 00          LOOP :FCB 0      ;MOST RIGHT POSITION
00FD 7F69 00          UD :FCB 0      ;up=00,down=FF
00FE 7F6A 00          COUNT :FCB 0      ;K(I) COUNTER (0 - LOOP)
00FF 7F6B 0000        FROM :FDB 0
0100 7F6D 0000        TO :FDB 0
0101 7F6F 00          XCT :FCB 0
0102 7F70 00          YCT :FCB 0
0103 7F71            CTRL :RMB 4      ;LPT CONTROL CODE AREA
0104 7F75            RCB :RMB 6
0105 7F7B            BUF1 :RMB 32      ;PATTERN DATA AREA(KANZI INPUT)
)
0106 7F9B            BUF2 :RMB 16      ;CHANGE DATA AREA (LPT OUTPUT)
0107 7FAB 0000        FDB 0      ;HSPACE DATA
0108 7FAD 0000        FDB 0
0109 7FAF 0000        FDB 0
010A 7FB1 0000        FDB 0
010B 7FB3 0000        FDB 0
010C 7FB5 0000        FDB 0
010D 7FB7 0000        FDB 0
010E 7FB9 00          FCB 0
010F 7FBA 3C          MOZISU:FCB 60      ;HSPACE=0
0110 7FBB 38          FCB 56
0111 7FBC 35          FCB 53
0112 7FBD 32          FCB 50
0113 7FBE 30          FCB 48
0114 7FBF 2D          FCB 45
0115 7FC0 2B          FCB 43
0116 7FC1 29          FCB 41
0117 7FC2 28          FCB 40
0118 7FC3 26          FCB 38
0119 7FC4 24          FCB 36
011A 7FC5 23          FCB 35
011B 7FC6 22          FCB 34
011C 7FC7 21          FCB 33

```

011D 7FCB 20  
011E 7FC9 1E  
011F 7FCA

FCB 32  
FCB 30  
END

; HSPACE=15

# MP-80 TYPE II を漢字プリンタに!



近藤 孝吉

FM-8を使っている方で、『漢字が普通のキャラクタと同じ大きさでプリンタに印字できたら』と思っている方も多いと思います。MP-80Kを持っている人は別ですが……。そこでMB27401もしくはMP-80 TYPE IIでも普通のキャラクタと同じ大きさで印字できるようにプログラムを作ってみました。

## 方法

富士通から発売されている日本語出力ライブラリVOL. 1またはHARDC 2によって漢字を印字した場合、どうしても縦方向が16ドット分必要なために字が大きくなります。MP-80Kの場合は紙送りがドットの半分で打てるので、縦方向を圧縮して印字できるのです。それではMP-80 TYPE IIでは(MB27401も同じ)紙送りが1ドットなのでできないかと言うと、なんとマニュアルにはないのですがESC + "J" + n というコマンドがあって、ドットの半分で紙送りができるのです。

つまりESC + "L" と ESC + "J" を組み合わせて、MP-80 K とまったく同じように印字できるのです。ただし、MP-80 K では18×14ですが、これが16×16となり少し字体が違いますが大きさは普通のキャラクタ並になります。

プログラムは日本語出力ライブラリの出力ライブラリを参考にして作りました。ポジション・インディペンデントなプログラムになっており、どこへ置いても走りますが、リストはディスク使用の方の場合も考えて5000H 番地からなっています。プログラム自身が必要なのは5000H ~ 5164

H までで、51D7H まではワーク・エリアです。

## 使い方

整数変数に漢字コードを入れて U S R 関数で5000H を呼べばよく、53字を越えるかCHR\$(13) が送られると印字して改行します。その例をBASICのプログラムに示します。文字は16×16ドットで文字間隔は2ドットです。

なお、漢字プリントを終了する場合は必ずESC + "A" + 12を実行してください。これは紙送りをプログラム中で変更して11ドット+1/2ドット+1/2ドットの計12ドットとして印字しているため、そのままにしておくと紙送りが11ドットのままになるので改行ピッチが1/6インチとなくなります。

以上で特に説明はいらないと思います。BASICのプログラムを参考にしていただければ良いと思います。ただし、漢字ROMが実装されていないと使えません。

### 参考文献

- 1) 日本語出力ライブラリVOL. 1, 富士通
- 2) MP-80K, 説明書
- 3) MP-80TYPE II, 説明書
- 4) 富士通 スカイラブ・シャトル創刊号
- 5) アプソリュート・アセンブラ解説書, 富士通

### サンプル印字例(原寸)

I/O 工学社  
FM-8 漢字プリントプログラム

### サンプル・プログラム

```
10 CLEAR, &H4FFF
20 DEF USR1=&H5000
30 OPEN "O", 1, "LPT0:"
40 READ A%
50 IF A%=0 THEN PRINT#1, CHR$(&H1B)+ "A" + CHR$(12); END
60 A%=USR1(A%)
70 GOTO 40
100 DATA &H2349, &H213F, &H234F, &H2120, &H3929, &H3358, &H3C52, 13
102 DATA &H2346, &H234D, &H215D, &H2338, &H2120, &H3441, &H3B7A, &H2557, &H256A
104 DATA &H2573, &H2548, &H2557, &H256D, &H2530, &H2569, &H2560, 13, 0
```

### アセンブル・リスト

00010		NAM	KANPRI	00120	500E A6	8D 0155	LDA	SCOUNT, PCR
00020		OPT	SYM, NOOBJ	00130	5012 30	8D 0152	LEAX	BUFF, PCR
00030	5000	ORG	*5000	00140	5016 EF	86	STU	A, X
00040		006A	RYTES	00150	5018 4C		INCA	
00050		5000	START	00160	5019 4C		INCA	
00060	5000 81	02	CMPL	00170	501A A7	8D 0149	STA	SCOUNT, PCR
00070	5002 26	1E	5022	00180	501E 81	6A	CMPL	#BYTES
00080	5004 EC	02	LDD	00190	5020 24	01 5023	BCC	MAIN
00090	5006 10B3	000D	CMPL	00200	5022 39		RETURN	RTS
00100	500A 27	17	5023	00202	5023 E6	8D 0140	LDB	SCOUNT, PCR
00110	500C 1E	03	EXG	00204	5027 86	09	LDA	#*09



## アセンブル・リスト

00206	5029	3D				MUL				00710	50D9	6A	8D	00FA		DEC		PEND4,PCR
00208	502A	30	8D	0129		LEAX		POPEN4,PCR		00720	50DD	26	F4	50D3		BNE		SHIF3
00210	502E	E7	84			STB		,X		00730	50DF	35	10			PULS		X
00212	5030	A7	01			STA		1,X		00740	50E1	39				RTS		
00214	5032	30	8D	012F		LEAX		POPEN8,PCR		00750	50F2	30	8D	0079	PRINT1	LEAX		POPEN7,PCR
00216	5036	E7	84			STB		,X		00760	50E6	AF	8D	0071		STX		POPEN6,PCR
00218	5038	A7	01			STA		1,X		00770	50FA	30	8D	006B		LEAX		POPEN5,PCR
00220	503A	30	8D	0114		LEAX		POPEN3,PCR		00780	50EE	AD	9F	FBFA		JSR		[#FBFA]
00222	503E	AF	8D	010C		STX		POPEN2,PCR		00790	50F2	39				RTS		
00230	5042	30	8D	0106		LEAX		POPEN1,PCR		00800	50FC	30	8D	0051	PRINT2	LEAX		PDAT8,PCR
00240	5046	AD	9F	FBFA		JSR		[#FBFA]		00810	50F7	AF	8D	0049		STX		PDAT7,PCR
00259	504A	A6	8D	0119		LDA		SCOUNT,PCR		00820	50FB	30	8D	0043		LEAX		PDAT6,PCR
00260	504E	30	8D	0116		LEAX		BUFF,PCR		00830	50FF	AD	9F	FBFA		JSR		[#FBFA]
00270	5052	30	86			LEAX		A,X		00832	5103	39				RTS		
00280	5054	AF	8D	017A		STX		PEND1,PCR		00840	5104		16		KDAT1	FCB		\$16,
00290	5058	30	8D	00AE		LEAX		KDAT4,PCR			5105		00					
00300	505C	AF	8D	0174		STX		PEND2,PCR		00850	5106		0002		KDAT2	RMB		2
00310	5060	33	8D	0104	MAIN1	LEAU		BUFF,PCR		00860	5108		0002		KDAT3	RMB		2
00320	5064	11A3	8D	0169	MAIN2	CMPU		PEND1,PCR		00870	510A		0002		KDAT4	RMB		2
00330	5069	27	36	50A1		BEQ		MAIN3		00880	510C		001E		KDAT5	RMB		30
00340	506B	EC	C1			LDD		,U++		00890	512A		0E		PDAT1	FCB		\$0E,
00350	506D	ED	8D	0097		STD		KDAT3,PCR			512B		00					
00360	5071	30	8D	0095		LEAX		KDAT4,PCR		00900	512C		0002		PDAT2	RMB		2
00370	5075	AF	8D	00BD		STX		KDAT2,PCR		00910	512E		0012		PDAT22	FDB		\$0012
00380	5079	30	8D	00B7		LEAX		KDAT1,PCR		00940	5130		0010		PDAT4	RMB		16
00390	507D	AD	9F	FBFA		JSR		[#FBFA]		00942	5140		0000			FDB		\$0000
00400	5081	AE	8D	014F		LDX		PEND2,PCR		00950	5142		0E		PDAT6	FCB		\$0E,
00410	5085	31	8D	00A7		LEAY		PDAT4,PCR			5143		00					
00420	5089	8D	2F	50BA		BSR		SHIFT		00952	5144		0002		PDAT7	RMB		2
00430	508B	30	01			LEAX		\$01,X		00954	5146		0004			FDB		\$0004
00440	508D	8D	2B	50BA		BSR		SHIFT		00956	5148		1B4A		PDAT8	FDB		\$1B4A
00450	508F	30	8D	009D		LEAX		PDAT4,PCR		00958	514A		01			FCB		1
00460	5093	AF	8D	0095		STX		PDAT2,PCR		00960	514B		0A			FCB		\$0A
00470	5097	30	8D	00BF		LEAX		PDAT1,PCR		00970	514C		0E		POPEN1	FCB		\$0E,
00480	509B	AD	9F	FBFA		JSR		[#FBFA]			514D		00					
00490	509F	20	C3	5064		BRA		MAIN2		00980	514E		0002		POPEN2	RMB		2
00500	50A1	30	8D	0067	MAIN3	LEAX		KDAT5,PCR		00990	5150		0007			FDB		\$0007
00510	50A5	AC	8D	012B		CMFX		PEND2,PCR		01000	5152		1B41		POPEN3	FDB		\$1B41
00520	50A9	26	07	50B2		BNE		MAIN4		01010	5154		0B			FCB		11
00530	50AB	8D	46	50F3		BSR		PRINT2		01020	5155		1B4C			FDB		\$1B4C
00540	50AD	6F	8D	00B6		CLR		SCOUNT,PCR		01030	5157		0002		POPEN4	RMB		2
00550	50B1	39				RTS				01040	5159		0E		POPEN5	FCB		\$0E,
00560	50B2	AF	8D	011E	MAIN4	STX		PEND2,PCR			515A		00					
00570	50B6	8D	2A	50E2		BSR		PRINT1		01050	515B		0002		POPEN6	RMB		2
00580	50B8	20	A6	5060		BRA		MAIN1		01060	515D		000B			FDB		\$000B
00590	50BA	86	0B			LDA		#0B		01070	515F		0D		POPEN7	FCB		\$0D
00600	50BC	A7	8D	0116		STA		PEND3,PCR		01080	5160		1B4A			FDB		\$1B4A
00610	50C0	8D	09	50CB	SHIF1	BSR		SHIF2		01090	5162		01			FCB		1
00620	50C2	31	21			LEAY		\$01,Y		01100	5163		1B4C			FDB		\$1B4C
00630	50C4	6A	8D	010E		DEC		PEND3,PCR		01102	5165		0002		POPEN8	RMB		2
00640	50C8	26	F6	50C0		BNE		SHIF1		01110	5167		0001		SCOUNT	RMB		1
00650	50CA	39				RTS				01120	5168		006A		BUFF	RMB		BYTES
00660	50CB	86	0B			LDA		#0B		01130	51D2		0002		PEND1	RMB		2
00670	50CD	A7	8D	0106		STA		PEND4,PCR		01140	51D4		0002		PEND2	RMB		2
00680	50D1	34	10			PSHS		X		01150	51D6		0001		PEND3	RMB		1
00690	50D3	6B	B1		SHIF3	ASL		,X++		01160	51D7		0001		PEND4	RMB		1
00700	50D5	69	A4			ROL		,Y		01170			5000			END		START
00702	50D7	30	02			LEAX		\$02,X										



## 濃淡スクリーン・コピー

■松本忠博

I/O'82年4月号p.208, BIG I/OプラザにFM-8用の「カラーノウタン・プリント」というプログラムが載っていました。今回、これとほぼ同じ動作をするプログラムをマシン語で作成したので発表します。

それでは4月号のプログラムとの相違点について順に述べます。

- ①6809のマシン語でできています。もちろんポジション・インディペンデントです。(0, 0) から(639, 199)までの濃淡処理に要する時間は30秒弱です。ただし、本プログラムにはHARDC2は含まれていません。
- ②オリジナルの方は濃淡をつけるために「ランスウニョル ドット ノ ショウキョ」を行っていますが、

プログラムでは乱数は使わず、(x, y)座標の数値から算術的に消去すべき点を決定しています(その方が美しいと判断したため)。

- ③①で述べたようにHARDC2は実行されないのでも本プログラム実行後、自分でHARDC2を実行する必要があります。

- ④高速化のため処理は、ドット単位ではなく横1列ずつ処理(GET, PUT)しています。

プログラム中いくつかむだな部分もありますが、自分なりにわかりやすく作ったつもりです。ROMのバージョンにも依存していないと思います。なお、ワーク・エリアとして13Dから240バイト使っています。細かいところはアセンブル・リストを見てもらった方が早いでしょう。

## 濃淡スクリーン・コピーアセンブル・リスト

```

10000
10001
10002
10003
10004
10005
10006
10007
10008
10009
10010      (6000)
10011
10012      (00C5)
10013      (00C7)
10014      (00F5)
10015      (043D)
10016
10017
10018
10019      6000      1A 50
10020      6002      8D 29
10021      6004      32 7E
10022      6006      8D 61
10023      6008      8D 2A
10024      600A      86 50
10025      600C      A7 61
10026      600E      86 08
10027      6010      A7 E4
10028      6012      8D 28
10029      6014      8D 42
10030      6016      6A E4
10031      6018      26 F8
10032      601A      30 01
10033      601C      6A 61
10034      601E      26 EE
10035      6020      17 009D
10036      6023      0C C8
10037      6025      96 C8
10038      6027      81 C8
10039      6029      26 DB
10040      602B      35 90
10041
10042
10043
10044      602D      5F 4F
10045      602F      DD C5
10046      6031      DD C7
10047      6033      39
10048
10049
10050
10051      6034      8E 043D
10052      6037      0F C5
10053      6039      0F C6
10054      603B      39
10055
10056
10057
10058      603C      D6 C8
10059      603E      86 05
10060      6040      3D

*****
** SHADE ROUTINE **
*****
** IF ((X+Y*5) MOD 8) >= CL AND CL <> 7 THEN PRESET(X,
**   X : 0-639
**   Y : 0-199
**
** ORG      $6000
**
** X0      EQU    $C5
** Y0      EQU    $C7
** MOD     EQU    $F5
** BUFF    EQU    $43D
**
** MAIN
**
** START   ORCC    ##50      :MASK IRQ & FIRO
**          BSR    INIT1
**          LEAS    -2,S
** LOOP1    BSR    GET      :GET@A(0,y)-(639,y),BUFF,G
**          BSR    INIT2
**          LDA     #80
**          STA     1,S
** LOOP2    LDA     #8
**          STA     S
** LOOP3    BSR    CALCUL
**          BSR    CARRY
**          DEC     ,S
**          BNE     LOOP3
**          LEAX    1,X
**          DEC     1,S
**          BNE     LOOP2
**          LBSR    PUT      :PUT@ (0,y)-(639,y),BUFF,PSET
**          INC     Y0+1
**          LDA     Y0+1
**          CMPA    #200
**          BNE     LOOP1    :Repeat, until y=200
**          PULS    X,PC      :RETURN.
**
** INITIALIZE 1
**
** INIT1    CLRD
**          STD     X0
**          STD     Y0
**          RTS
**
** INITIALIZE 2
**
** INIT2    LDX     #BUFF
**          CLR     X0
**          CLR     X0+1
**          RTS
**
** CALCULATE "MOD"
**
** CALCUL   LDB     Y0+1
**          LDA     #5
**          MUL

```

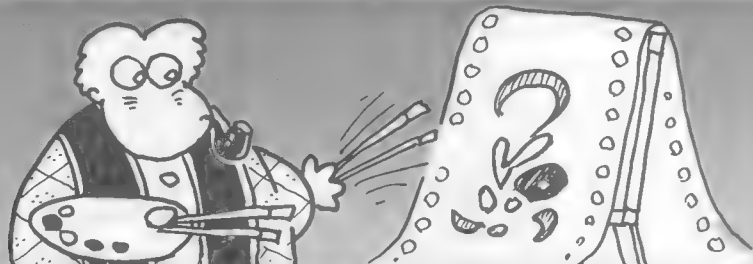
## アセンブル・リスト

```

10061 6041 DB C6          ADDB X0+1
10062 6043 C4 07          ANDB #7
10063 6045 D7 F5          STB MOD
10064 6047 0C C6          INC X0+1
10065 6049 39          RTS
10066
10067 * GET COLOR CODE
10068 *
10069 604A 4F          COLOR CLRA
10070 604B 68 B9 00A0 ASL 160,X
10071 604F 49          ROLA
10072 6050 68 B8 50 ASL 80,X
10073 6053 49          ROLA
10074 6054 68 B4 ASL ,X
10075 6056 49          ROLA
10076 6057 39          RTS
10077
10078 * SET ON CARRY
10079 *
10080 605B 8D F0          CARRY BSR COLOR
10081 605A 5F          CLRB
10082 605B 81 07          CMPA #7
10083 605D 27 05          BEQ RESET
10084 605F 90 F5          SUBA MOD
10085 6061 22 01          BHI RESET
10086 6063 5C          SET INCB
10087 6064 EA B4          RESET ORB ,X
10088 6066 E7 B4          STB ,X
10089 6068 39          RTS
10090
10091 * GET ONE LINE in BUFFER
10092 *
10093 6069 8D 41          GET BSR HALT
10094 606B 86 1D          LDA #1D :GET BLOCK 2
10095 606D 8D 28          BSR SETXY
10096 606F 8D 4B          BSR SUB
10097 6071 CE 043D          LDU #BUFF
10098 6074 8D 10          BSR GETTFR
10099 6076 86 64          LDA #100
10100 607B 87 FC82          STA #FC82
10101 607B 8D 3F          BSR SUB
10102 607D 8D 07          BSR GETTFR
10103 607F 86 B0          LDA #80
10104 6081 B7 FC80          STA #FC80
10105 6084 20 36          BRA SUB
10106
10107 6086 8E FC83          GETTFR LDX #FC83
10108 6089 8D 21          BSR HALT
10109 608B E6 B0          LDB ,X+
10110 608D 27 07          BEQ G2
10111 608F A6 B0          G1 LDA ,X+
10112 6091 A7 C0          STA ,U+
10113 6093 5A          DECB
10114 6094 26 F9          BNE G1
10115 6096 39          G2 RTS
10116
10117 * SET X,Y
10118 *
10119 6097 8E FC82          SETXY LDX #FC82
10120 609A A7 B0          STA ,X+
10121 609C 5F 4F          CLRD
10122 609E ED B4          STD ,X :X1
10123 60A0 CC 027F          LDD #139
10124 60A3 ED 04          STD 4,X :X2
10125 60A5 DC C7          LDD Y0
10126 60A7 ED 02          STD 2,X :Y1
10127 60A9 ED 06          STD 6,X :Y2
10128 60AB 39          RTS
10129
10130 * HALT SUB-MPU
10131 *
10132 60AC B6 FD05          HALT LDA #FD05
10133 60AF 2B FB          BMI HALT
10134 60B1 B6 B0          LDA #80
10135 60B3 B7 FD05          STA #FD05
10136 60B6 B6 FD05          WAIT LDA #FD05
10137 60B9 2A FB          BFL WAIT
10138 60BB 39          RTS
10139
10140 * RESTART SUB-MPU
10141 *
10142 60BC 7F FD05          SUB CLR #FD05
10143 60BF 39          RTS
10144
10145 * PUT ONE LINE
10146 *
10147 60C0 8D EA          PUT BSR HALT
10148 60C2 86 1C          LDA #1C :PUT BLOCK 1
10149 60C4 8D D1          BSR SETXY
10150 60C6 10BE 043D          LDY #BUFF
10151 60CA 8E FC8B          LDX #FC8B
10152 60CD CC 0003          LDD #0003 :AND
10153 60D0 ED B1          STD ,X++
10154 60D2 C6 50          LDB #80
10155 60D4 E7 B0          STB ,X+
10156 60D6 54          LSRB
10157 60D7 EE A1          P1 LDU ,Y++
10158 60D9 EF B1          STU ,X++
10159 60DB 5A          DECB
10160 60DC 26 F9          BNE F1
10161 60DE 20 DC          BRA SUB
10162
10163 * END
(60DF)

```

# カラーコピー



OECU EDPS JG3ECZ

各社からカラープリンタが発表され始めました。そこで、カラーリボンを使って専用プリンタ (MP-80type II) によるカラーハード・コピーを考えてみました。

## はじめに

現在、市販されているMP-80用カラーカートリッジ・リボンは黒、赤、緑、紫、茶の5色が出ているようです。これを使えばリボンの交換により擬似的にカラーコピーが可能になります。ただし、色は2色少なく、画面と同じ色が出せないという欠点があります。コピーの大きさはHARD C1では大きすぎて荒くなり、HARDC2では横長になってしまいます。そこで、HARDC1の半分の大きさ、つまり画面1ドットに対して、ブリタンは横2ドット、縦1/2ドット、倍密度ビット・イメージにしました。

## 使い方

BASICからUSR命令により実行します。引数は整数型で図1のようになっており、"1"にしたビットに対応する色がプリントされます。したがって単色コピーの場合、引数は&HFEとなりますが、0のときもこれと同じになるようにしてあります。&HFFの場合、何も実行しません。また、8ビット目は"0"で左側、"1"で右側にプリントされます。一枚の用紙に4回コピーできます。

プログラムをリスト1のように変更することにより、コピーモードは画面1ビットに対し、横1ドット、縦1/2ドット、標準ビット・イメージになります。マシン語はポジション・インディペンデントですが、プログラムの後に320バイトのバッファ領域が必要なので、DISK BASICの場合、&H6D00から入れてください。

# カラーコピーのアイデア

カラーコピーを行なうに当たって、最大の問題は重ね打ちの際に位置を合わせることです。これは用紙1枚をプリンタに通し、上と下をつなぎ合わせてループ状にすればフォーム・フィードによりピッタリと位置が揃います(写真1)。

写真 1

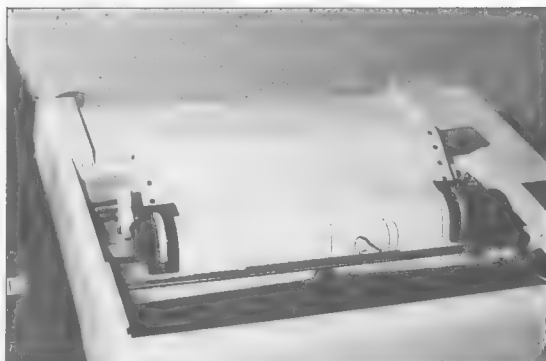
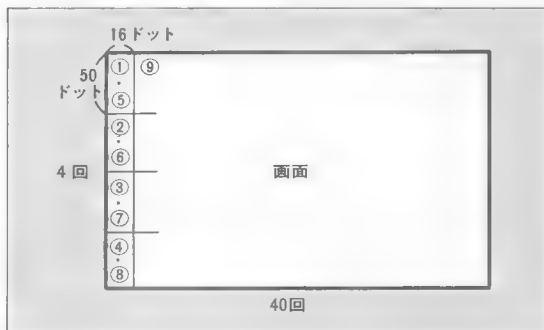


图 1 引数

15								8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	位置	白	黄	水色	緑	紫	赤	青	黒

図2 GET BLOCK1によるパター入力



です。実行はプログラム1のように行ないます。

## プログラムの説明

BIOSのサブシステム・インプット・ルーチン(SUBIN)のうち、GET BLOCKIコマンドにより、画面データを読み取り、プリンタに出力します。メイン部とサブ部の共有メモリは128バイトしかないため、X方向16ドット、Y方向50ドット計100バイトずつ読み取ります(図2)。こうすればX

## リスト 1

7E0E 02-01  
7E39 02-01  
7EB8 4C-4B

## プログラム 1

```
10 CLEAR, &H7CFF: LOADM "HARDC3"
20 DEFINT A-Z: DEFUSR = &H7D00
30 OPEN "O", 1, "LPT0": PRINT #1, CHR$(12);
40 '
50 ' クラフィック サクセイ
60 '
70 FOR I=0 TO 6
80   A=2^I: A=USR(A): COPY
90   PRINT #1, CHR$(12);: ' フォームフィート"
100  A$=INPUT$(1): ' リボン コウカン
110 NEXT I
120 END
```

方向の16ドットがちょうど2バイトのデータとなります。このうち偶数または奇数ビットめを取り出し、バッファに1行分のデータ200バイトをたくわえます。

プリンタに打ち出す際、高速化のため“1”のある最も左のデータから、“1”のある最も右のデータまでをビット・イメージでプリントしています。左側は6ビットをスペース1文字に置き換えることにより、ヘッドは最短距離を動くことになります。このプログラムはプリンタにはデータを1バイトずつ出力しているため、プリンタが打ち終わるまでCPUは次のルーチンへ行きません。プリンタ用バッファを400バイトにし、データを一気に出力することにより、プリンタ動作中にCPUは次のルーチンの処理をするため、少し早くなるでしょう。

## 最後に

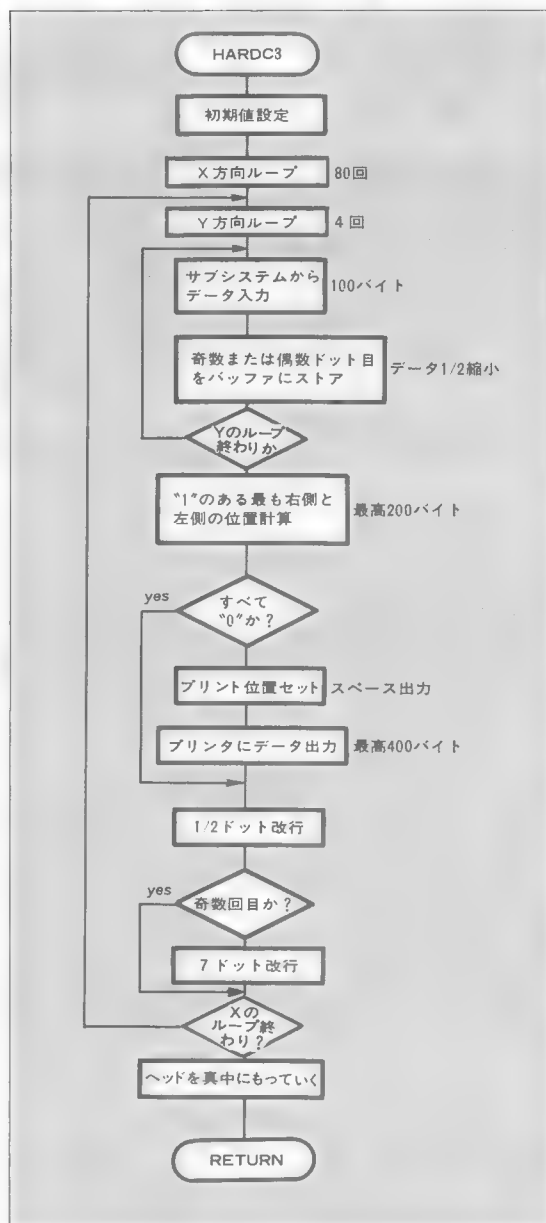
実行時間は最高2分51秒、最低41秒です。HARDC1の4分37秒、HARDC2の2分10秒に比べると高速であり、単色コピーとしては有効です。カラーコピーとして使う場合、用紙をループ状にする、リボンを何度も交換するという過酷な使い方をするため、プリンタの故障のもとであり、インク・リボンのムダ使いでもあり、多用することは避けるほうがよいでしょう。カラープリンタが安く手に入る日が待ち遠しいものです。

最後に写真を撮ってくれた中村君、ありがとう。

## 参考文献

“FM-8活用研究”，工学社

## フローチャート



## カラーコピー プログラム・リスト

```

0001 0000 ;
0002 0000 ; COLOR HARD COPY
0003 0000 ;
0004 0000 ; program by J63ECZ/3
0005 0000 ; 1982/10/20
0006 0000 ; position independent program
0007 0000 ;
0008 0000 ;
0009 7D00 ORG $7D00 ;diskBASIC is $6D00 ;P3
000A 7D00 ; initialize
000B 7D00 EC02 LDD 2,X ;argument load
000C 7D02 C1FF CMPB #$FF ;if COL=$FF then return
000D 7D04 2704 BEQ *+6
000E 7D06 8102 CMPA #2 ;if POS>=2 then return
000F 7D08 2501 BCS *+3
0010 7D0A 39 RTS
0011 7D0B 5D TSTB
0012 7D0C 2602 BNE *+4
```

## カラーコピー プログラム・リスト

```

0013 7D0E C6FE      LDB  ##FE
0014 7D10 30BD01AD  LEAX RCB,PC      ;X=RCB
0015 7D14 E71C      STB  -4,X      ;COL
0016 7D16 C605      LDB  #5      ;L position
0017 7D18 4D        TSTA
0018 7D19 2702      BEQ  *+4
0019 7D1B C62B      LDB  #43     ;R position
001A 7D1D E71B      STB  -5,X      ;POS
001B 7D1F B60E      LDA  ##0E     ;printer out code
001C 7D21 A7B4      STA  ,X
001D 7D23 31B8ED    LEAY -19,X      ;Y=ESCA
001E 7D26 10AF02    STY  2,X
001F 7D29 CC0003    LDD  #3
0020 7D2C ED04      STD  4,X
0021 7D2E BDF2D8    JSR  BIOS
0022 7D31 6A05      DEC  5,X
0023 7D33 31B8E6    LEAY -26,X      ;Y=DC2
0024 7D36 10AF02    STY  2,X
0025 7D39 BDF2D8    JSR  BIOS
0026 7D3C           ; XLOOP set
0027 7D3C 6F1D      CLR  -3,X      ;XLOOP clear      ;P4
0028 7D3E           ; YLOOP set
0029 7D3E 6F8D017D  LOOP1:CLR  YLOOP,PC  ;YLOOP clear
002A 7D42           ; sub system input
002B 7D42 30BD0183  LOOP2:LEAX INBUF,PC  ;X=INBUF
002C 7D46 861B      LDA  ##1B     ;GET BLOCK2 sub command code
002D 7D48 A702      STA  2,X
002E 7D4A A615      LDA  -11,X     ;XLOOP
002F 7D4C 44        LSRA
0030 7D4D C610      LDB  #16
0031 7D4F 3D        MUL
0032 7D50 ED03      STD  3,X      ;X1 set
0033 7D52 CB0F      ADDB #15
0034 7D54 ED07      STD  7,X      ;X2 set
0035 7D56 A616      LDA  -10,X     ;YLOOP
0036 7D58 C632      LDB  #50
0037 7D5A 3D        MUL
0038 7D5B ED05      STD  5,X      ;Y1 set
0039 7D5D CB31      ADDB #49
003A 7D5F ED09      STD  9,X      ;Y2 set
003B 7D61 6F0B      CLR  11,X     ;color number clear      ;P5
003C 7D63 A614      LDA  -12,X     ;COL
003D 7D65 310C      LEAY 12,X     ;Y=color code address
003E 7D67 5F        CLRB
003F 7D6B 44        LSRA
0040 7D69 2404      BCC  *+6
0041 7D6B E7A0      STB  ,Y+
0042 7D6D 6C0B      INC  11,X     ;color code set,address inc
0043 7D6F 5C        INCB      ;color number set
0044 7D70 4D        TSTA
0045 7D71 26F5      BNE  *-9
0046 7D73 AF1A      STX  -6,X
0047 7D75 8611      LDA  ##11     ;subsystem input code
0048 7D77 A71B      STA  -8,X     ;RCB
0049 7D79 CC0068    LDD  #104
004A 7D7C ED1E      STD  -2,X
004B 7D7E C60C      LDB  #12
004C 7D80 EB0B      ADDB 11,X
004D 7D82 ED1C      STD  -4,X
004E 7D84 301B      LEAX -8,X      ;X=RCB
004F 7D86 BDF2D8    JSR  BIOS
0050 7D89           ; printer buffer set
0051 7D89 300C      LEAX 12,X      ;X=INBUF+4      ;P6
0052 7D8B 8604      LDA  #4
0053 7D8D A012      SUBA -14,X     ;YLOOP
0054 7D8F C632      LDB  #50
0055 7D91 3D        MUL
0056 7D92 31BD019A  LEAY OUTBUF-1,PC  ;Y=OUTBUF-1
0057 7D96 31AB      LEAY D,Y
0058 7D98 8632      LDA  #50
0059 7D9A E68D0120  LOOP3:LDB  XLOOP,PC
005A 7D9E C401      ANDB #1
005B 7DA0 2704      BEQ  *+6      ;if odd then 1dot L shift
005C 7DA2 6901      ROL  1,X
005D 7DA4 6984      ROL  ,X
005E 7DA6 C608      LDB  #8
005F 7DAB 6684      ROR  ,X      ;input buffer 2dot R shift
0060 7DAA 6601      ROR  1,X
0061 7DAC 6684      ROR  ,X
0062 7DAE 6601      ROR  1,X

```

## カラーコピー プログラム・リスト

```

0063 7DB0 66A4 ROR ,Y ;output buffer 1dot R shift
0064 7DB2 5A DECB
0065 7DB3 26F3 BNE *-11
0066 7DB5 3002 LEAX 2,X
0067 7DB7 313F LEAY -1,Y
0068 7DB9 4A DECA
0069 7DBA 26DE BNE LOOP3
006A 7DBC ; YLOOP count
006B 7DBC 308D00FF LEAX YLOOP,PC ;X=YLOOP ;P7
006C 7DC0 6C84 INC ,X
006D 7DC2 A684 LDA ,X
006E 7DC4 8104 CMPA #4
006F 7DC6 1026FF7B LBNE LOOP2
0070 7DCA ; output byte length
0071 7DCA 318D0162 LEAY OUTBUF-1,PC ;Y=OUTBUF-1
0072 7DCE CC00CB LDD #200
0073 7DD1 6DAB TST D,Y
0074 7DD3 2605 BNE *+7
0075 7DD5 5A DECB ; B: data last address
0076 7DD6 26F9 BNE *-5
0077 7DD8 206B BRA JAMP ;if all clear then jump
0078 7DDA 3402 PSHS A ;(S)=0 (S):space number
0079 7DDC 3121 LEAY 1,Y ;Y=OUTBUF
007A 7DDE 6DA0 TST ,Y+ ;,buffer next address
007B 7DE0 260E BNE *+16
007C 7DE2 4C INCA
007D 7DE3 8106 CMPA #6
007E 7DE5 26F7 BNE *-7
007F 7DE7 A7E2 STA ,-S
0080 7DE9 E0E0 SUBB ,S+
0081 7DEB 4F CLRA
0082 7DEC 6CE4 INC ,S ;space number increment
0083 7DEE 20EE BRA *-16
0084 7DF0 ;print position set
0085 7DF0 308D00CD LEAX RCB,PC ;X=RCB ;P8
0086 7DF4 311A LEAY -6,X ;SPACE
0087 7DF6 10AF02 BTY 2,X
0088 7DF9 860E LDA #$0E ;printer out code
0089 7DFB A784 STA ,X
008A 7DFD 8601 LDA #1
008B 7DFF A705 STA 5,X
008C 7E01 A6E4 LDA ,S ;space number load
008D 7E03 AB1B ADDA -5,X ;POS
008E 7E05 BDF2D8 JSR BIOS
008F 7E08 4A DECA
0090 7E09 26FA BNE *-4
0091 7E0B ; bit image set
0092 7E0B 3404 PSHS B
0093 7E0D 8602 LDA #2 ; or #1
0094 7E0F 3D MUL ;bit image data number
0095 7E10 1E89 EXG A,B
0096 7E12 ED18 STD -8,X ;ESCL+2
0097 7E14 3504 PULS B
0098 7E16 8604 LDA #4
0099 7E18 A705 STA 5,X
009A 7E1A 3116 LEAY -10,X ;Y=ESCL
009B 7E1C 10AF02 STY 2,X
009C 7E1F BDF2D8 JSR BIOS
009D 7E22 ; data out
009E 7E22 8601 LDA #1 ; ;P9
009F 7E24 A705 STA 5,X
00A0 7E26 318D0107 LEAY OUTBUF,PC ;Y=OUTBUF
00A1 7E2A 3502 PULS A ;space number load
00A2 7E2C 3404 PSHS B
00A3 7E2E C606 LDB #6
00A4 7E30 3D MUL
00A5 7E31 31AB LEAY D,Y ;Y=data out 1st address
00A6 7E33 3504 PULS B
00A7 7E35 10AF02 STY 2,X
00A8 7E38 8602 LDA #2 ; or #1
00A9 7E3A BDF2D8 JSR BIOS ;data out , 2or1 loop
00AA 7E3D 4A DECA
00AB 7E3E 26FA BNE *-4
00AC 7E40 3121 LEAY 1,Y ;buffer next address
00AD 7E42 5A DECB
00AE 7E43 26F0 BNE *-14
00AF 7E45 ; printer LF
00B0 7E45 308C79 JAMP :LEAX <RCB,PC ;X=RCB
00B1 7E48 860E LDA #$0E ;printer out code
00B2 7E4A A784 STA ,X

```



```

00B3 7E4C 318C5A      LEAY <ESCJ,PC      ;Y=ESCJ
00B4 7E4F 10AF02      STY 2,X
00B5 7E52 C604        LDB #4
00B6 7E54 A61D        LDA -3,X      ;XLOOP
00B7 7E56 B401        ANDA #1
00B8 7E58 2701        BEQ *+3      ;if odd then printer LF
00B9 7E5A 5C          INCB
00BA 7E5B 4F          CLRA
00BB 7E5C ED04        STD 4,X
00BC 7E5E BDF2D8      JSR BIOS
00BD 7E61              ; XLOOP count
00BE 7E61 6C1D        INC -3,X      ;XLOOP
00BF 7E63 A61D        LDA -3,X
00C0 7E65 B150        CMPA #80
00C1 7E67 1026FED3    LBNE LOOP1
00C2 7E6B              ; printer normalize
00C3 7E6B 311A        LEAY -6,X      ;Y=SPACE      ;P10
00C4 7E6D 10AF02      STY 2,X
00C5 7E70 B601        LDA #1
00C6 7E72 A705        STA 5,X
00C7 7E74 B628        LDA #40      ;head center position
00C8 7E76 BDF2D8      JSR BIOS
00C9 7E79 4A          DECA
00CA 7E7A 26FA        BNE *-4
00CB 7E7C 3138        LEAY -8,Y      ;ESCK
00CC 7E7E 10AF02      STY 2,X
00CD 7E81 B604        LDA #4
00CE 7E83 A705        STA 5,X
00CF 7E85 BDF2D8      JSR BIOS      ;bit image 1dot out
00D0 7E8B 3123        LEAY 3,Y      ;Y=ESCK+3:null code address
00D1 7E8A 10AF02      STY 2,X
00D2 7E8D B601        LDA #1
00D3 7E8F A705        STA 5,X
00D4 7E91 BDF2D8      JSR BIOS
00D5 7E94 3136        LEAY -10,Y     ;Y=CR
00D6 7E96 10AF02      STY 2,X
00D7 7E99 BDF2D8      JSR BIOS
00D8 7E9C 3125        LEAY 5,Y      ;Y=ESC2
00D9 7E9E 10AF02      STY 2,X
00DA 7EA1 6C05        INC 5,X
00DB 7EA3 BDF2D8      JSR BIOS
00DC 7EA6 39          RTS
00DD 7EA7              ; work area
00DE 7EA7      BIOS :EQU $F2D8      ;
00DF 7EA7 12      DC2 :FCB $12      ;
00E0 7EAB 14      DC4 :FCB $14      ;DC2:head move speed normalize
00E1 7EA9 1B      ESCJ :FCB $1B      ;DC4:head move speed normalize
00E2 7EAA 4A          FCB $4A      ;ESC 1/2dot LF
00E3 7EAB 01          FCB $01      ;"J"
00E4 7EAC 0D      CR :FCB $0D      ;1
00E5 7EAD 0A      LF :FCB $0A      ;CR
00E6 7EAE 1B      ESCA :FCB $1B      ;LF
00E7 7EAF 41          FCB $41      ;ESC 7dot LF
00E8 7EB0 07          FCB $07      ;"A"
00E9 7EB1 1B      ESC2 :FCB $1B      ;7
00EA 7EB2 32          FCB $32      ;ESC 12dot LF
00EB 7EB3 1B      ESCK :FCB $1B      ;"2"
00EC 7EB4 4B          FCB $4B      ;ESC bit image code
00ED 7EB5 01          FCB $01      ;"K"
00EE 7EB6 00          FCB $00      ;1
00EF 7EB7 1B      ESCL :FCB $1B      ;0
00F0 7EB8 4C          FCB $4C      ;ESC bit image code
00F1 7EB9 00          FCB 0      ;"L" or #4B:"K"
00F2 7EBA 00          FCB 0      ;low
00F3 7EBB 20      SPACE :FCB $20      ;high
00F4 7EBC 00      POS :FCB 0      ;" "
00F5 7EBD 00      COL :FCB 0      ;position L:5,R:43
00F6 7EBE 00      XLOOP :FCB 0      ;color code $01-$FE
00F7 7EBF 00      YLOOP :FCB 0      ;80 count
00F8 7EC0 00      NUL :FCB 0      ;4 count
00F9 7EC1          RCB :RMB 8      ;not use
00FA 7EC9          INBUF :RMB 104      ;subsystem input buffer
00FB 7F31          OUTBUF:RMB 200      ;printer output buffer
00FC 7FF9          ;
00FD 7FF9          ; END
00FE 7FF9          ;

```

## FMオシロ

FM-8を

低周波オシロの代用にする

(単位なし)

このプログラムの特徴はハードの追加なしに、プログラムのみでFM-8が低周波オシロスコープになるということです。この機能を使うにはBASICでは遅すぎるので、アナログ・ポートからの読み込みはマシン語を使っています。

FM-8内蔵のシリアル型A/Dコンバータではこれが限界だと思います。“GALAXY FLY”のように、サブCPUにプログラムを転送してそのプログラムでサブCPUが制御されるので、表示スピードが速くなりました。また、時間軸の表示間隔が自由に選択でき、経過時間を測定スピードと表示時間から計算して、たとえば12msecというように、表示できるようになっています。

## 使い方

図1のように、DIN5ピンオス型コネクタ（70円ぐらい）を買ってきて、FM-8裏のアナログ・コネクタに差し込み、ラジオやテープレコーダーやアンプのイヤホン端子から、みの虫クリップ付の接続コードでDIN5ピンソケットのG端子（金枠部分）にマイナス、1ピンにプラス側を接続すればOKです。

入力が交流の場合、マイナス側がカットされてしまいます。全部の波形を出したい場合は図2の回路で入力にバイアス電圧をかけます。若干ハムがはいりますが、

BASICプログラムの930行のTIG20#=TIG#-7の7を大きい数字にすると、経過時間線のきめが細くなり、小さくすると荒くなります。

**STOP** キーを押すと止まりますが、サブCPUのキャラクタ・バッファにプログラムを書き込んでいるため、たまに“READY”が出なくなるときがありますが、そのときは**STOP** キーを押しながらリセット・ボタンを押し、その後モニター・モードでアドレス01E1に\$CC、01E2に\$57をいれるともとの状態に戻ります。

1秒間の測定回数および測定スピードの有効数字は3桁ぐらいです。

サンプル印字

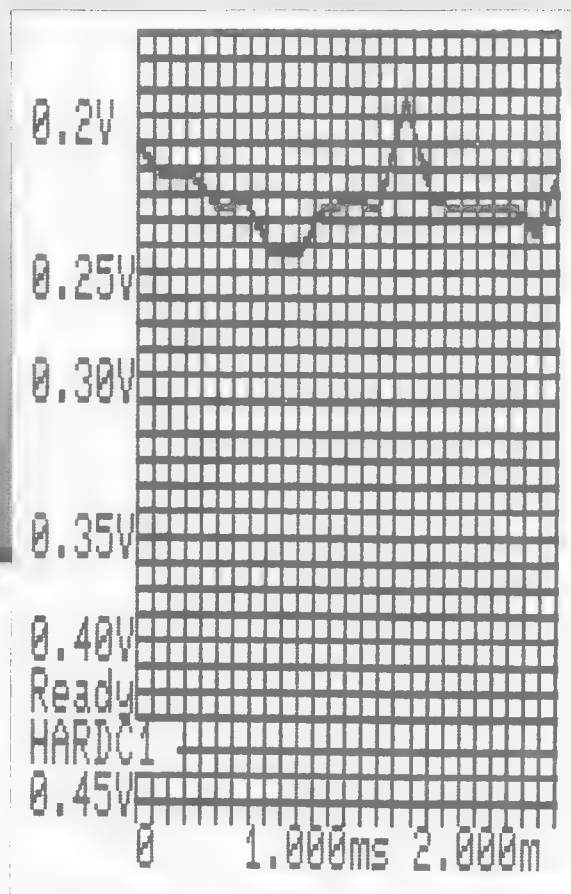


図1 接続図

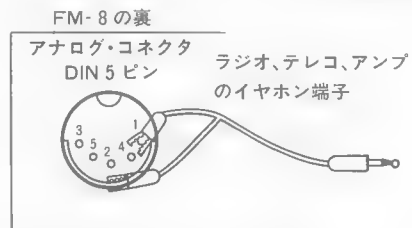
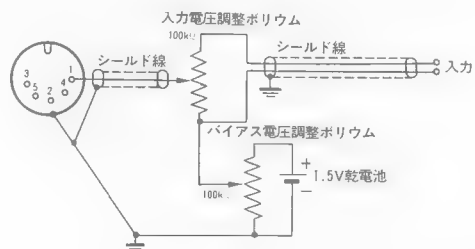


図2 バイアスをかけるときの接続図



## BASICプログラム・リスト

```

10 COLOR 4: CLEAR, &H2FFF: WIDTH 80, 25: CONSOLE 0, 25, 0, 0: PRINT " FM-B ショ"
20 PRINT " マシンコ" フロク ラム ハ ロート スミテスカ ? Y OR N"
30 L$=INKEY$
40 IF ( L$ = "Y" ) OR ( L$ = "y" ) THEN 130
50 IF ( L$ = "N" ) OR ( L$ = "n" ) THEN 70
60 GOTO 30
70 PRINT " カセット ロート" シュンビ OK ? Y OR N "
80 L1$=INKEY$
90 IF ( L1$ = "Y" ) OR ( L$ = "y" ) THEN 120
100 IF ( L1$ = "N" ) OR ( L$ = "n" ) THEN 80
110 GOTO 80
120 LOADM"OSHIROM"
130 PRINT:PRINT:PRINT:PRINT " ニュウリョク レンシ" ハ 5/8 V テス (ラシ"オヤ テレコ ノ イヤホン タン
シ カラ ヨウセツ ニュウリョク シテモ タ"イシ"ョフ テス )" :PRINT:PRINT:PRINT"STOP KEY ラ オスト トマリマス":PR
INT:PRINT:PRINT:PRINT:
140 PRINT"アナロク" ホート ヲ ニュウリョク ノ シカタ ハ , DIN 5 ヒン オス カタ コネクタ (¥70) ラ カッテキテ FM-
B ノ ウラ ノ アナロク" コネクタ ニ マシコミ , G タンシ(カナワク フ"フ"ン) ニ マイナス , 1 ヒンニ フ"ラス ラ ニュウリョクシマス
。"
150 PRINT:PRINT:PRINT:PRINT"ツキ" ニ スンテ" イイテスカ ? Y or N "
160 A$=INKEY$
170 IF ( A$="Y" ) OR ( L$="y" ) THEN 180 ELSE 160
180 CLS:LOCATE 0,2
190 PRINT" 1カ"ン ノ ソクテイ カイズ ムミコミ ヒット ヒョウシ" ト"ット ソクテイ スピード" INP
UT NUMBER "
200 PRINT " 8 ヒット 0-199 1-
8"
210 PRINT " ショウイ 7 ヒット 0-199 1-
7"
220 PRINT " ショウイ 6 ヒット 0-199 1-
6"
230 PRINT " ショウイ 5 ヒット 0-199 1-
5"
240 PRINT " ショウイ 4 ヒット 0-199 1-
4"
250 PRINT " ショウイ 3 ヒット 0-199 1-
3"
260 PRINT " ショウイ 2 ヒット 0-199 1-
2"
270 PRINT " ショウイ 1 ヒット 0-199 1-
1"
280 PRINT " カイ 7 ヒット 0-125 1- -
7"
290 PRINT " カイ 6 ヒット 0- 63 1- -
6"
300 PRINT " カイ 5 ヒット 0- 31 1- -
5"
310 PRINT " カイ 4 ヒット 0- 15 1- -
4"
320 PRINT " カイ 3 ヒット 0- 7 1- -
3"
330 PRINT " カイ 2 ヒット 0- 3 1- -
2"
340 PRINT " カイ 1 ヒット 0- 1 1- -
1"
350 Y=3:FOR S=8 TO 1 STEP -1:SP1#=(1/((18+20*S)/1079000#))%1:Y=Y+1:LOCATE 6,Y:PR
INT USING "#####";SP1#;:PRINT" カイ":NEXT
360 Y=3:FOR S=8 TO 1 STEP -1:SP1#=(18+20*S)/1.079#:Y=Y+1:LOCATE 50,Y:PRINT USING
"#####";SP1#;:PRINT" マイクロ":NEXT
370 Y=11:FOR S=7 TO 1 STEP -1:SP1#=(1/((18+8*(8-S)+20*S)/1079000#))%1:Y=Y+1:LOCA
TE 6,Y:PRINT USING "#####";SP1#;:PRINT" カイ":NEXT
380 Y=11:FOR S=7 TO 1 STEP -1:SP1#=(18+8*(8-S)+20*S)/1.079#:Y=Y+1:LOCATE 50,Y:PR
INT USING "#####";SP1#;:PRINT" マイクロ":NEXT
390 CONNECT(0,14)-(639,14)-(639,155)-(0,155)-(0,14)
400 LINE (142,14)-(142,155),OR
410 LINE(0,28)-(639,28),OR
420 LINE (265,14)-(265,155),OR
430 LINE (380,14)-(380,155),OR
440 LINE (455+64,14)-(455+64,155),OR
450 PRINT:PRINT:PRINT:INPUT"INPUT NUMBER = ";A:IF (A>8) OR (A=0) THEN 450
470 INPUT"ヒョシ" カンカ? ??? ト"ット ニ 1カイ ソクテイ"ン ヒョウシ" (1-255)";H
480 IF (H <=0) OR (H>255) THEN 470
490 H1=FIX(599/H)+1:IF H1>126 THEN H1=126
495 IF H1 <2 THEN H2 =2ELSE H2=H1
500 POKE &H6926,H:POKE&H6944,H2-1
510 CLS:COLOR 3:LOCATE 45,21:PRINTH;"ト"ット ニ 1 カイ ソクテイ テン ヒョウシ"
520 LOCATE 0,0
530 PRINT"0.00V":PRINT:PRINT"0.05V":PRINT
540 PRINT:PRINT"0.10V"
550 PRINT
560 PRINT"0.15V"
570 PRINT
580 PRINT
590 PRINT"0.2V"
600 PRINT
610 PRINT
620 PRINT"0.25V"
630 PRINT
640 PRINT"0.30V"
650 PRINT
660 PRINT
670 PRINT"0.35V"
680 PRINT:PRINT"0.40V"

```

## BASICプログラム・リスト

```

690 PRINT
700 PRINT
710 PRINT"0.45V"
720 FOR Y=0 TO 46 :Y1=4.08*Y:Y2=0:Y2=Y1-FIX(Y1)-.5:IF Y2 >0 THEN Y1=FIX(Y1)+1
730 LINE (40,Y1)-(439,Y1),OR,1:NEXT
740 FOR Y=0 TO 46 STEP 5 :Y1=Y*4.08:Y2=0:Y2=Y1-FIX(Y1)-.5:IF Y2 >0 THEN Y1=FIX(Y1)
)+1
750 LINE (40,Y1)-(439,Y1),OR,2:NEXT
760 IF A=8 THEN POKE &H61F3,H1+1:BIT=8:GOSUB1020:GOSUB920:EXEC&H61DE:GOTO1070
770 IF A=-7 THEN POKE &H5B52,H1+1:BIT=7:GOSUB1030:GOSUB920:EXEC&H5B3D:GOTO1070
780 IF A=-6 THEN POKE &H5BFD,H1+1:BIT=6:GOSUB1030:GOSUB920:EXEC&H5BEB:GOTO1070
790 IF A=-5 THEN POKE &H5CAB,H1+1:BIT=5:GOSUB1030:GOSUB920:EXEC&H5C93:GOTO1070
800 IF A=-4 THEN POKE &H5D53,H1+1:BIT=4:GOSUB1030:GOSUB920:EXEC&H5D3E:GOTO1070
810 IF A=-3 THEN POKE &H5DFE,H1+1:BIT=3:GOSUB1030:GOSUB920:EXEC&H5DE9:GOTO1070
820 IF A=-2 THEN POKE &H5EA9,H1+1:BIT=2:GOSUB1020:GOSUB920:EXEC&H5E94:GOTO1070
830 IF A=-1 THEN POKE &H5F56,H1+1:BIT=1:GOSUB1020:GOSUB920:EXEC&H5F41:GOTO1070
840 IF A=5 THEN POKE &H6D3F,H1+1:BIT=5:GOSUB1020:GOSUB920:EXEC&H6D2A:GOTO1070
850 IF A=4 THEN POKE &H6DFO,H1+1:BIT=4:GOSUB1020:GOSUB920:EXEC&H6DD8:GOTO1070
860 IF A=3 THEN POKE &H6EA1,H1+1:BIT=3:GOSUB1020:GOSUB920:EXEC&H6EBC:GOTO1070
870 IF A=2 THEN POKE &H6F5B,H1+1:BIT=2:GOSUB1020:GOSUB920:EXEC&H6F43:GOTO1070
880 IF A=1 THEN POKE &H6C84,H1+1:BIT=1:GOSUB1020:GOSUB920:EXEC&H6C6F:GOTO1070
890 IF A=-2 THEN POKE &H6BD4,H1+1:BIT=2:GOSUB1030:GOSUB920:EXEC&H6BFF:GOTO1070
900 IF A=-1 THEN POKE &H6B27,H1+1:BIT=1:GOSUB1030:GOSUB920:EXEC&H6B12:GOTO1070
910 GOTO 180
920 TIG#=SP** (H1):IF H=0 THEN LOCATE 5,25 :PRINT"0";:RETURN: 'PRINT"GAMEN NO KEY
KA GIKAN":TIG#;
930 TIG20#=TIG#/7:' 1/7 GAMEN NO KEIKS GIKAN
940 MOB=LEN(STR$(FIX(TIG20#)))-1:'PRINT"MOB-1=";MOB;
950 TIGG#=(TIG20#*(10^(MOB-1)))*(10^(MOB-1)):IF TIGG#=0 THEN GOSUB 1040 'PRINT"
1/7 GAMEN NO KEIKA GIKAN K":TIGG#;
960 SEN#=(TIGG#/10)/(SP#/H):'PRINT"SEN NO KANKAKU":SEN#;
962 IF (SEN#<3.5) AND (SEN#>=1.5) THEN SEN#=3*SEN#:TIGG#=TIGG#*3
964 IF (SEN#<1.5) AND (SEN#>=.5) THEN SEN#=4*SEN#:TIGG#=TIGG#*4
966 IF (SEN#<.5) THEN SEN#=8*SEN#:TIGG#=TIGG#*8
970 FOR X=40 TO 639 STEP SEN#:LINE (X,0)-(X,191),OR,1:NEXT
980 FOR X=40 TO 639 STEP10* SEN#:LINE (X,0)-(X,191),OR,2:NEXT
990 LOCATE 5,25:PRINT"0";
1000 T#=0:FOR L=10*SEN#/8+2 TO 69 STEP 10*SEN#/8:LOCATE L,25:T#=T#+TIGG#:PRINT U
SING"###.###":T#;:PRINT"msec";:NEXT
1010 RETURN
1020 SP#=(18+20*BIT)/1079#:RETURN
1030 SP#=(18+8*(B-BIT)+20*BIT)/1079#:RETURN
1040 IF FIX(TIG20#*100)>0 THEN TIGG#=((TIG20#*100)*1)/10:RETURN
1050 IF FIX(TIG20#*100)>0 THEN TIGG#=((TIG20#*100)*1)/100:RETURN
1060 PRINT"ER":PRINT"TIG20#=";TIG20#:STDP
1070 FOR X=0 TO 100:A =0 :NEXTX:LOCATE 20,20:END

```

## マシン語ダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5800	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	:0E
5810	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	:0E
5820	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	:0E
5830	97	0C	0F	0C	D6	0C	E7	80	C6	88	D7	0C	C6	00	D7	0C	:E1
5840	97	0C	D7	0C	97	0C	D7	0C	D6	0C	E7	80	97	0C	0F	0C	:13
5850	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	:0E
5860	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	:0E
5870	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	:0E
5880	D6	0C	E7	80	C6	88	D7	0C	C6	00	D7	0C	97	0C	D7	0C	:A9
5890	97	0C	D7	0C	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	:D6
58A0	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	:0E
58B0	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	:0E
58C0	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	:0E
58D0	C6	88	D7	0C	C6	00	D7	0C	97	0C	D7	0C	97	0C	D7	0C	:E6
58E0	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	:0E
58F0	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	:0E

Sum: 19 3C 58 78 C6 30 0B 60 48 30 58 78 58 B4 30 EC :F3

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5900	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	:0E
5910	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	C6	88	D7	0C	:B1
5920	C6	00	D7	0C	97	0C	D7	0C	97	0C	D7	0C	D6	0C	E7	80	:FE
5930	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	:0E
5940	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	:0E
5950	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	:0E
5960	97	0C	0F	0C	D6	0C	E7	80	C6	88	D7	0C	C6	00	D7	0C	:E1
5970	97	0C	D7	0C	97	0C	D7	0C	D6	0C	E7	80	97	0C	0F	0C	:13
5980	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	:0E
5990	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	:0E
59A0	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	:0E
59B0	D6	0C	E7	80	39	00	00	00	00	00	00	00	00	00	00	00	:82
59C0	C6	88	D7	0C	C6	04	D7	0C	97	0C	0F	0C	97	0C	0F	0C	:5A
59D0	97	0C	0F	0C	97	0C	0F	0C	97	0C	0F	0C	97	0C	0F	0C	:F8
59E0	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	:0E
59F0	97	0C	0F	0C	D6	0C	E7	80	39	00	00	00	00	C6	88	D7	:65

Sum: 48 30 58 78 BB AC 49 6C 24 24 72 60 DB DE 22 C3 :1C

5A00	0C	97	0C	0F	0C	97	0C	0F	0C	97	0C	0F	0C	97	0C	0F	:F8
5A50	0C	97	0C	0F <td>0C</td> <td>97</td> <td>0C</td> <td>0F</td> <td>0C</td> <td>97</td> <td>0C</td> <td>0F</td> <td>0C</td> <td>97</td> <td>0C</td> <td>0F</td> <td>:F8</td>	0C	97	0C	0F	0C	97	0C	0F	0C	97	0C	0F	:F8
5A60	0C	D6	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	39	00	00	:95
5A70	00	00	C6	88	D7	0C	C6	00	D7	0C	97	0C	D7	0C	97	0C	:03
5A80	D7	0C	97	0C	D7	0C	97	0C	D7	0C	97	0C	D7	0C	97	0C	:18
5A90	D7	0C	97	0C	D7	0C	97	0C	D7	0C	97	0C	D6	0C	E7	80	:97
5AA0	0F	0C	D6	0C	E7	80	39	00	00	00	C6	88	D7	0C	C6	04	:D8
5AB0	D7	0C	97	0C	0F	0C	97	0C	0F	0C	97	0C	0F	0C	97	0C	:C0
5AC0	0F	0C	97	0C	0F	0C	97	0C	0F	0C	97	0C	0F	0C	97	0C	:F8
5AD0	0F	0C	97	0C	0F	0C	D6	0C	E7	80	39	00	00	00	00	C6	:21
5AE0	88	D7	0C	C6	00	D7	0C	97	0C	D7	0C	97	0C	D7	0C	97	:B7
5AF0	0C	D7	0C	97	0C	D7	0C	97	0C	D7	0C	97	0C	D7	0C	97	:18

Sum: 9A 03 EF AE 55 39 91 54 EA 38 19 63 DE 5C 11 E6 :7C

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5B00	0C	D7	0C	97	0C	D7	0C	97	0C	D7	0C	D6	0C	E7	80	39	:7D
5B10	00	00	00	7F	5B	80	3B	86	7F	4A	26	FD	96	05	2B	FC	:F9
5B20	1A	50	86	80	97	05	C6	88	5A	27	FD	96	05	2A	FC	86	:9F
5B30	02	B7	FC	80	86	00	97	05	1C	AF	35	12	3B	34	7F	BD	:14
5B40	60	17	BE	5B	13	BF	01	E1	B6	BA	B7	5B	80	86	FD	1F	:B8
5B50	8B	86	7F	BE	34	00	C6	24	CE	63	48	10	AE	C1	10	AF	:66
5B60	81	5A	26	F7	4A	26	FE	86	39	A7	84	BE	30	00	86	80	:05
5B70	BD	34	00	86	1B	4A	26	FD	96	05	2B	FC	1A	50	86	80	:2E
5B80	97	05	C6	88	5A	27	FD	96	05	2A	FC	86	01	87	FC	80	:63
5B90	BE	30	00	CE	FC	B1	C6	07	6F	C4	64	80	69	C4	5A	26	:9A
5BA0	F9	33	41	11	83	FD	00	26	ED	86	00	97	05	1C	AF	20	:1E
5BB0	00	BE	CC	57	BF	01	E1	7E	61	8B	39	00	00	00	7F	5C	:D0
5BC0	5B	3B	86	7F	4A	26	FD	96	05	2B	FC	1A	50	86	80	97	:D1
5BD0	05	C6	88	5A	27	FD	96	05	2A	FC	86	02	B7	FC	80	86	:53
5BE0	00	97	05	1C	AF	35	12	3B	34	7F	BD	60	17	BE	5B	86	:77
5BF0	BF	01	E1	B6	BA	B7	5C	5B	86	FD	1F	8B	86	00	7F	5C	:43

Sum: BE 98 0B 35 9F 70 25 24 CF 62 0C 14 9D 07 AC 77 :D3

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5C00	00	C6	22	CE	63	DF	10	AE	C1	10	AF	B1	5A	26	F7	4A	:78
5C10	26	EF	86	39	A7	84	BE	30	00	86	80	80	34	00	86	1B	:52
5C20	4A	26	FD	96	05	2B	FC	1A	50	86	80	97	05	C6	88	5A	:63
5C30	27	FD	96	05	2A	FC	86	01	B7	FC	80	9E	30	00	0C	FC	:27
5C40	B1	C6	06	6F	C4	64	80	69	C4	5A	26	F9	33	41	11	83	:12
5C50	FD	00	26	ED	86	00	97	05	1C	AF	20	00	8E	C8	57	BF	:8D
5C60	01	E1	7E	61	8B	39	00	00	7F	5D	06	3B	86	7F	4A	27	:1E
5C70	26	FD	96	05	2B	FC	1A	50	86	80	97	05	C6	88	5A	26	:9A
5C80	FD	96	05	2A	FC	86	02	B7	FC	80	86	00	97	05	1C	AF	:66

SC90	35	12	38	34	7F	BD	60	17	BE	5C	69	BF	01	E1	B6	BA	:9D
SCA0	B7	5D	06	B6	FD	1F	BB	86	7F	BE	34	00	C6	20	CE	64	:26
SCB0	68	10	AE	C1	10	AF	B1	5A	26	F7	4A	26	EF	86	39	A7	:66
SCC0	B4	BE	30	00	B6	BD	34	00	B6	18	4A	26	FD	96	05	:DF	
SCD0	2B	FC	1A	50	B6	80	97	05	C6	08	5A	27	FD	96	05	:A4	
SCE0	FC	B6	01	B7	FC	B0	BE	30	00	CE	FC	B1	C6	05	6F	:C4	
SCF0	64	80	69	CA	5A	26	F9	33	41	11	B3	FD	00	26	ED	:B6	
Sum:	9F	21	23	D4	50	DA	9A	01	64	EE	C7	3B	BB	D1	34	:5B	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5D00	00	97	05	1C	AF	20	00	8E	CC	57	BF	01	E1	7E	61	BB	:70
5D10	39	00	00	00	7F	5D	B1	3B	86	7F	4A	26	FD	96	05	2B	:39
5D20	FC	1A	50	B6	80	97	05	C6	08	5A	27	FD	96	05	2A	FC	:15
5D30	86	02	B7	FC	B0	BE	30	00	97	05	1C	AF	35	12	3B	34	:7D
5D40	BD	60	17	BE	5D	1A	BF	01	E1	B6	BA	B7	5D	81	B6	7F	:5C
5D50	1F	BB	86	7F	BE	34	00	C6	1E	CE	64	EF	10	AE	C1	10	:05
5D60	AF	B1	5A	26	F7	4A	26	EF	86	39	A7	84	BE	30	00	86	:34
5D70	80	BD	34	00	B6	18	4A	26	FD	96	05	2B	FC	1A	50	86	:2E
5D80	80	97	05	C6	08	5A	27	FD	96	05	2A	FC	86	01	B7	FC	:63
5D90	80	BE	30	00	CE	FC	B1	C6	04	6F	CA	64	80	69	C4	5A	:F1
5DA0	26	F9	33	41	11	B3	FD	00	26	ED	B6	00	97	05	1C	AF	:24
5DB0	20	00	8E	CC	57	BF	01	E1	7E	61	BB	39	00	00	00	7F	:00
5DC0	5E	5C	B8	66	7F	4A	26	FD	96	05	2B	FC	1A	50	B6	80	:99
5DD0	97	05	C6	08	5A	27	FD	96	05	2A	FC	86	02	B7	FC	80	:64
5DE0	86	00	97	05	1C	AF	35	12	3B	34	7F	BD	60	17	BE	5D	:5D
5DF0	BF	BF	01	E1	B6	BA	B7	5E	5C	96	FD	1F	BB	86	7F	8E	:41
Sum:	46	1A	C6	1B	4F	B6	9A	A9	51	1A	7B	A5	21	10	B1	E6	:A6

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5E00	34	00	C6	1C	CE	59	FD	10	AE	C1	10	AF	B1	5A	26	F7	70
5E10	4A	26	EF	86	39	A7	84	BE	30	00	B6	80	BD	34	00	B6	84
5E20	18	4A	26	FD	96	05	2B	FC	1A	50	B6	80	97	05	C6	0B	84
5E30	5A	27	FD	96	05	2A	FC	86	01	B7	FC	B0	BE	30	00	CE	85
5E40	FC	B1	C6	03	6F	C4	64	80	69	C4	5A	26	F9	33	41	11	8B
5E50	83	FD	00	26	ED	B6	00	97	05	1C	AF	20	00	8E	CC	57	51
5E60	BF	01	E1	7E	61	BB	39	00	00	00	7F	5F	09	3B	86	7F	93
5E70	4A	26	FD	96	05	2B	FC	1A	50	B6	80	97	05	C6	08	5A	84
5E80	27	FD	96	05	2A	FC	86	02	B7	FC	B0	BE	30	97	05	1C	DE
5E90	AF	35	F7	39	34	7F	BD	60	17	BE	5E	6A	BF	01	E1	B6	00
5EA0	BB	B7	5F	09	3B	86	7F	1F	BB	86	03	BE	34	00	C6	22	CE
5EB0	65	A9	10	AE	C1	10	AF	B1	5A	26	F7	4A	26	EF	86	39	85
5EC0	A7	84	BE	30	00	B6	80	BD	34	00	B6	18	4A	26	FD	96	81
5ED0	05	2B	FC	1A	50	B6	80	97	05	C6	08	5A	27	FD	96	05	1F
5EE0	2A	FC	86	01	B7	FC	B0	BE	30	00	CE	FC	B1	C6	07	6F	25
5EF0	CA	64	80	69	C4	5A	26	F9	33	41	11	B3	FD	00	00	00	7F
Sum:	05	DD	90	1B	D4	46	F8	9A	36	6B	12	8B	52	3E	AC	47	F7

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5F00	26	EF	86	39	A7	05	1C	AF	20	00	8E	CC	57	BF	01	:E1	:70
5F10	7E	5E	6E	39	00	00	00	7F	5F	09	3B	86	7F	4A	:26	:FD	:C6
5F20	96	05	2B	FC	1A	50	B6	80	97	05	:C6	08	5A	27	FD	96	:80
5F30	05	2A	FC	86	02	B7	FC	B0	BE	30	:00	97	05	:1C	AF	35	:87
5F40	34	7F	BD	60	17	BE	5F	1F	BB	86	B7	5F	:17	BD	60	17	:87
5F50	BB	86	FD	1F	BB	86	7F	BE	34	00	C6	1E	CE	66	3D	10	:11
5F60	AE	C1	10	AF	B1	5A	26	F7	4A	26	EF	86	39	A7	8E	4F	:8E
5F70	30	00	B6	80	BD	34	00	B6	1E	4A	26	FD	96	05	2B	FC	:FA
5F80	1A	50	B6	80	97	05	:C6	08	5A	27	FD	96	05	2A	FC	B6	:9F
5F90	01	B7	FC	B0	BE	30	:00	CE	FC	B1	C6	06	6F	CA	64	80	:20
5FA0	69	CA	5A	26	F9	33	:41	11	B3	FD	00	26	:29	BD	60	17	:87
5FB0	E9	B6	00	97	05	1C	AF	20	00	8E	CC	57	BF	01	E1	7E	:C6
5FC0	5F	18	39	00	00	00	00	00	00	00	00	00	00	00	00	00	:B3
5FD0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
5FE0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
5FF0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
Sum:	DA	5F	42	B3	FF	F0	0A	F6	69	55	D2	E5	25	93	3D	96	:E1

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6000	BE	CC	57	BF	01	E1	35	7F	39	00	00	00	00	00	00	00	:3F
6010	00	00	00	00	00	00	00	00	B6	FD	05	2B	FC	1A	50	86	:0A
6020	B7	FD	05	BD	74	7F	FD	05	BE	00	00	30	1F	26	FC	B6	:F6
6030	FD	05	2B	FC	B6	80	B7	FD	05	17	00	91	7F	FD	05	BE	:9E
6040	00	00	00	30	1F	26	FC	BD	05	2B	FC	BE	68	02	BE	05	:05
6050	68	39	12	12	12	12	C6	FF	86	80	B7	FD	05	B7	FD	05	:29
6060	A6	00	B7	FC	B1	F7	FC	B0	7F	FD	05	7F	FD	05	B6	FD	:C2
6070	05	2B	FC	B6	80	B7	FD	05	B7	FD	05	B6	FC	BD	26	E8	:E3
6080	11	B3	64	E7	26	D2	B6	01	B7	FC	B0	7F	FD	05	7F	FD	:94
6090	05	BE	00	00	30	1F	26	FC	39	10	BE	FC	B0	30	BD	00	:19
60A0	0A	C6	21	A6	B0	A7	AO	5A	26	F9	39	00	00	3F	59	41	:E4
60B0	4D	41	55	43	48	49	93	D3	BF	90	34	03	1A	50	E7	D4	:6B
60C0	0A	B6	D4	0B	B6	D4	09	B6	D4	0A	35	83	10	BE	FC	00	:1D
60D0	80	30	BD	00	0A	C6	41	A6	B0	A7	AO	5A	26	F9	39	00	:6B
60E0	00	3F	59	41	4D	41	55	43	48	49	91	D3	96	3E	81	00	:49
60F0	2C	93	3E	81	90	34	7F	CE	C0	00	12	7F	D3	80	7F	D3	:85
Sum:	7B	C2	53	94	EF	8C	5B	4F	8B	50	DA	69	44	1A	55	1D	:37

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6100	80	7F	D3	80	B6	D4	0A	B6	D3	80	27	F8	B1	01	:27		
6110	B7	D4	0A	B7	D4	0A	B6	D3	B1	A7	C0	20	DB	7E	C0	00	:D4
6120	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
6130	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
6140	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
6150	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
6160	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00

6170	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
6180	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
6190	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
61A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
61B0	00	00	00	00	7F	62	51	3B	86	7F	4A	26	FD	96	05	2B :A5
61C0	FC	1A	50	B6	80	97	05	C6	08	5A	27	FD	96	05	2A :F5	
61D0	B6	02	B7	FC	B0	BE	30	00	97	05	1C	AF	35	7F	39	7A :F4
61E0	BD	60	17	BE	61	B4	BF	01	E1	B6	BA	B7	62	51	B6	FD :A5
61F0	1F	BB	86	3D	BE	34	00	C6	26	CE	62	5F	10	AE	01	10 :39

		+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6200	AF	B1	5A	26	F7	4A	26	EF	86	39	A7	84	BE	30	00	:B6	:3A	
6210	80	BD	34	00	B6	18	4A	26	FD	96	05	2B	FC	1A	50	86	:2E	
6220	80	97	05	C6	08	5A	27	FD	96	05	2A	FC	B6	01	B7	FC	:63	
6230	80	BE	30	00	CE	FC	B1	C6	08	6F	C4	64	80	69	C4	5A	:F5	
6240	26	F9	33	41	11	B3	FD	00	26	ED	B6	00	97	05	1C	AF	:24	
6250	20	00	8E	CC	57	BF	01	E1	7E	61	BB	39	00	0F	0C	C6	:2F	
6260	88	D7	0C	C6	00	D7	0C	97	0C	D7	0C	97	0C	D7	0C	D6	:F6	
6270	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	:0E	
6280	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	:0E	
6290	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	:0E	
62A0	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	39	B8	D7	0C	C6	:E4	
62B0	88	D7	0C	C6	00	D7	0C	97	0C	0F	0C	97	0C	0F	0C	D6	:66	
62C0	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	:0E	
62D0	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	:0E	
62E0	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	97	0C	0F	0C	D6	:0E	
62F0	0C	E7	80	97	0C	0F	0C	D6	0C	E7	80	39	00	00	00	C6	:79	

## マシン語ダンプ・リスト

```

6650 80 97 0C 0F 0C D6 0C E7 80 97 0C 0F 0C D6 0C E7 :0E
6660 80 97 0C 0F 0C D6 0C E7 80 97 0C 0F 0C D6 0C E7 :0E
6670 80 97 0C 0F 0C D6 0C E7 80 97 0C 0F 0C D6 0C E7 :80
6680 80 39 04 D7 0C C6 8B D7 0C C6 04 D7 0C 97 0C 0F :30
6690 0C 97 0C 0F 0C D6 0C E7 80 97 0C 0F 0C D6 0C E7 :9A
66A0 80 97 0C 0F 0C D6 0C E7 80 97 0C 0F 0C D6 0C E7 :0E
66B0 80 97 0C 0F 0C D6 0C E7 80 97 0C 0F 0C D6 0C E7 :E5
66C0 0C D6 0C E7 80 39 00 00 00 D6 8B D7 0C C6 00 D7 :5C
66D0 0C 97 0C D7 0C 97 0C D7 0C D6 0C E7 80 97 0C :13
66E0 0C D6 0C E7 80 97 0C 0F 0C D6 0C E7 80 97 0C :0E
66F0 0C D6 0C E7 80 97 0C 0F 0C D6 0C E7 80 39 0C :E8

```

Sum: D4 A0 AC D0 1C B7 30 A9 3C 71 2C 70 90 A7 30 F0 :3C

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6700 80 97 0C 0F 0C D6 0C E7 80 39 00 00 00 C6 8B D7 :E5
6710 0C C6 04 D7 0C 97 0C 0F 0C 97 0C 0F 0C D6 0C E7 :FE
6720 80 97 0C 0F 0C D6 0C E7 80 97 0C 0F 0C D6 0C E7 :0E
6730 80 97 0C 0F 0C D6 0C E7 80 39 0C 0F 0C D6 0C E7 :80
6740 80 97 0C 0F 0C D6 0B E7 80 39 00 00 00 C6 8B D7 :E1
6750 0C C6 00 D7 0C 97 0C D7 0C 97 0C D7 0C D6 0C E7 :8A
6760 80 97 0C 0F 0C D6 0C E7 80 97 0C 0F 0C D6 0C E7 :0E
6770 80 97 0C 0F 0C D6 0C E7 80 39 0C 0F 0C D6 0C E7 :80
6780 80 97 0C 0F 0C D6 0C E7 80 39 00 00 00 C6 8B D7 :E5
6790 0C C6 04 D7 0C 97 0C 0F 0C 97 0C 0F 0C D6 0C E7 :FE
67A0 80 97 0C 0F 0C D6 0C E7 80 97 0C 0F 0C D6 0C E7 :0E
67B0 80 39 0C E7 80 97 0C 0F 0C D6 0C E7 80 97 0C :E5
67C0 0C D6 0C E7 80 39 00 00 00 C6 8B D7 0C C6 00 D7 :5C
67D0 0C 97 0C D7 0C 97 0C D7 0C D6 0C E7 80 97 0C :13
67E0 0C D6 0C E7 80 97 0C 0F 0C D6 0C E7 80 39 0C :E8
67F0 80 97 0C 0F 0C D6 0C E7 80 97 0C 0F 0C D6 0C E7 :0E

```

Sum: 4B 1D A4 9B 1C 49 B0 09 CB 86 1B DB FB 05 2B 80 :A5

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6800 80 39 00 00 C6 8B D7 0C C6 04 D7 0C 97 0C 0F :E5
6810 97 0C 0F 0C D6 0C E7 80 97 0C 0F 0C D6 0C E7 :80
6820 39 0C 0F 0C D6 0C E7 0C 97 0C 0F 0C D6 0C E7 :25
6830 97 0C 0F 0C D6 0C E7 80 39 00 00 00 C6 8B D7 :01
6840 C6 00 D7 0C 97 0C D7 0C 97 0C D7 0C D6 0C E7 :FE
6850 97 0C 0F 0C D6 0C E7 80 39 0C 0F 0C 97 0C 0F :25
6860 97 0C 0F 0C D6 0C E7 80 97 0C 0F 0C D6 0C E7 :0E
6870 39 00 00 0C C6 8B D7 0C C6 04 D7 0C 97 0C 0F :05
6880 97 0C 0F 0C D6 0C E7 80 39 0C 0F 0C 97 0C 0F :25
6890 97 0C 0F 0C 97 0C 0F 0C 97 0C 0F 0C 97 0C 0F :FB
68A0 D6 0C E7 80 39 00 00 00 C6 8B D7 0C C6 00 D7 :5C
68B0 97 0C D7 0C 97 0C 0F 0C D6 0C E7 80 39 0C 0F :85
68C0 97 0C 0F 0C 97 0C 0F 0C 97 0C 0F 0C 97 0C 0F :FB
68D0 97 0C 0F 0C D6 0C E7 80 39 00 00 00 00 00 00 :40
68E0 7F D3 B0 E6 D4 0A B6 D3 80 27 FB B7 D4 0A 81 :01
68F0 27 16 B1 02 27 02 20 EE B6 D4 0A 8E C0 00 6F :C8

```

Sum: B3 A6 1D BC B7 A0 C9 15 CC F7 B1 49 3B 16 A3 ED :05

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6900 0C C2 E0 2D F9 35 7F 39 B6 D4 09 7F D3 B0 10 :9E
6910 D3 B1 CE CD 01 4F BE 00 2B AF C1 E6 A0 C1 C7 :25
6920 02 C6 C7 ED C1 C6 0A 3A BE 02 7F 2A 06 10 8C :DA
6930 00 25 E6 ED 00 00 BE 00 00 EF B1 8C BE 80 25 :F9
6940 CE CD 01 B6 3B 37 39 67 10 9F 67 30 9F 68 :53
6950 10 9F 6D 34 72 8D 0A 35 72 4A 26 8B B6 D4 :0A
6960 85 BE 00 01 CE FF FF 0C 68 93 67 9F 9D C3 00 :01
6970 DD 99 DD A3 DC 4D 93 69 2B 04 9F 9F 20 06 :DF
6980 DC 69 93 4D C3 00 01 DD 9B 9E 99 30 1F 10 :27
6990 80 9E 9B 30 1F 10 27 00 AF CC 00 00 DD A1 DC :67
69A0 DD 5A DC 69 DD 5B 8D 02 20 35 9E 5A 10 9E :5B
69B0 02 20 1E 34 32 CE 01 44 56 44 5A 1E 01 C4 :84
69C0 07 1E 02 B6 50 3D 33 8B CC E3 96 1E 02 E4 :A5
69D0 B2 34 04 E6 E4 33 C9 80 00 EA C4 E7 C4 35 :84
69E0 A1 D3 9B DD A1 93 99 23 0F DD A1 D6 59 DB :A7
69F0 59 BD 0C CA DC A1 20 ED 26 0B DD A1 D6 59 :DB

```

Sum: BF 24 2F 5A B4 72 3C 4A 9A FA FA F1 3B 5A 10 EB :F4

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6A00 D7 59 DC 5A D3 9D DD 5A 9E A3 30 1F 9F A3 :26
6A10 39 9E 67 10 9E 69 56 9E 2A 03 10 9E 6D 34 :32
6A20 61 44 56 44 56 44 56 1E 01 C4 07 1E 02 B6 :50
6A30 33 8B CC E3 96 02 E0 E6 A5 35 32 30 C4 BD :C0
6A40 33 8B 50 0A 9C 26 F4 39 9E 67 DE 6B 96 9D :2A
6A50 1E 13 9F 9D DF 9F 96 9E B4 07 BE E3 96 E6 :B8
6A60 5A 96 A0 B4 07 A6 B6 4A 43 34 06 DC 9D BD :63
6A70 04 DC 9F BD 5D E0 E0 26 25 E6 E0 E4 E0 9E :9F
6A80 9E 9D 10 9E 69 34 04 BD C0 D3 30 C4 31 E4 :30
6A90 80 00 5F A6 A5 A4 85 A7 B5 5A 2A F7 35 B4 :D7
6AA0 E6 E0 C1 FF 27 05 BD C1 9D 0A A3 E6 E0 2B :8D
6AB0 C1 0A 9E 9D 30 8D 9F 9D 0A A3 2B BB 10 9E :69
6AC0 9D BD C0 D3 33 C9 80 00 D6 A3 B6 FF A7 C5 :5A
6AD0 FB 39 44 56 44 56 44 56 39 C6 18 CE 5A 39 :10
6AE0 C1 10 AF B1 5A 26 F7 4A 7F 6B B5 3B 86 7F :4A
6AF0 FD 96 05 2B FC 1A 50 86 B0 97 05 C6 08 5A :27

```

Sum: 6E BC 19 FE 6E FD AB 2C F2 6C 1B 13 60 D0 72 BC :0D

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6B00 96 05 2A FC B6 02 B7 FC B0 86 00 97 05 1C :AF
6B10 12 3B 34 7F BD 60 17 BE 6A EB BF 01 E1 B6 :BA
6B20 6B B5 B6 FD 1F B8 B6 7F BE 34 00 C6 18 CE :5A
6B30 10 AE C1 10 AF B1 5A 26 F7 4A 26 EF B6 39 :A7

```

```

6B40 BE 30 00 B6 B0 BD 34 00 B6 1B 4A 26 FD 96 :05
6B50 FC 1A 50 B6 B0 97 05 C6 08 5A 27 FD 96 :05
6B60 B6 01 B7 FC B0 8E 30 00 CE FC B1 C6 01 :6F
6B70 B0 69 C4 5A 26 F9 33 41 11 83 FD 00 26 :ED
6B80 97 05 1C AF 20 00 BE CC 57 BF 01 E1 7E :61
6B90 00 00 00 00 00 7F 6C 32 3B 86 7F 4A 26 :FD
6BA0 2B FC 1A 50 B6 B0 97 05 C6 08 5A 27 FD :96
6BB0 FC B6 02 B7 FC B0 86 00 97 05 1C AF 35 :12
6BC0 7F BD 60 17 BE 6B 95 BF 01 E1 B6 BA B7 :6C
6BD0 FD 1F B8 B6 7F BE 34 00 C6 1A CE 5A 72 :10
6BE0 10 AF B1 5A 26 F7 4A 26 EF B6 39 A7 B4 :8E
6BF0 B6 B0 BD 34 00 B6 1B 4A 26 FD 96 05 2B :FC

```

Sum: B3 B9 D1 CB 8E 3C BC 6B A7 AD ED F7 EC AC 9B 0D :0E

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6C00 B6 B0 97 05 C6 08 5A 27 FD 96 05 2A FC :86
6C10 FC B0 8E 30 00 CE FC B1 C6 02 6F C4 64 :80
6C20 5A 26 F9 33 41 11 83 FD 00 26 ED B6 00 :97
6C30 AF 20 00 BE CC 57 BF 01 E1 7E 61 B8 39 :A9
6C40 00 00 00 00 00 7F 6C 32 3B 86 7F 4A 26 :FD
6C50 2B FC 1A 50 B6 B0 97 05 C6 08 5A 27 FD :96
6C60 FC B6 02 B7 FC B0 86 00 97 05 1C AF 35 :12
6C70 7F BD 60 17 BE 6C 45 BF 01 E1 B6 AC B7 :6C
6C80 FD 1F B8 B6 7F BE 34 00 C6 0A CE 68 AB :10
6C90 10 AF B1 5A 26 F7 4A 26 EF B6 39 A7 B4 :8E
6CA0 B6 B0 BD 34 00 B6 1B 4A 26 FD 96 05 2B :FC
6CB0 B6 B0 97 05 C6 08 5A 27 FD 96 05 2A FC :86
6CC0 FC B0 8E 30 00 CE FC B1 C6 01 6F C4 64 :80
6CD0 5A 26 F9 33 41 11 83 FD 00 26 ED B6 00 :97
6CE0 C4 33 41 11 83 FD 00 26 ED B6 00 97 05 :1C
6CF0 AC BE CC 57 BF 01 E1 7E 61 B8 39 00 00 :00

```

Sum: 10 BA BE 2D 54 70 F7 7E DF 7A 4B F9 2B D2 0A 94 :F3

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6D00 7F 6D A3 3B B6 7F 4A 26 FD 96 05 2B FC :1A
6D10 B0 97 05 C6 08 5A 27 FD 96 05 2A FC B6 :02
6D20 B0 86 00 97 05 1C AF 35 7F 39 34 7F BD :60
6D30 6D 00 BF 01 E1 B6 B4 B7 6D A3 86 FD 1F :B8
6D40 BE 34 00 C6 1A CE 66 39 10 AE C1 10 AF :B1
6D50 F7 4A 26 EF B6 39 A7 B4 BE 30 00 B6 B0 :BD
6D60 B6 1B 4A 26 FD 96 05 2B FC 1A 50 B6 B0 :97
6D70 0B 5A 27 FD 96 05 2A FC B6 01 B7 FC B0 :8E
6D80 CE FC B1 C6 05 6F C4 64 B0 69 C4 5A 26 :F9
6D90 6B C4 68 C4 33 41 11 83 FD 00 26 ED B6 :00
6DA0 1C AF 20 00 BE CC 57 BF 01 E1 7E 6D 04 :39
6DB0 00 7F 6E 56 3B B6 7F 4A 26 FD 96 05 2B :FC
6DC0 B6 B0 97 05 C6 08 5A 27 FD 96 05 2A FC :86
6DD0 FC B0 86 00 97 05 1C AF 35 7F 39 34 7F :BD
6DE0 BE 6D B1 BF 01 E1 B6 B2 B7 6E 56 B6 FD :1F
6DF0 7F BE 34 00 C6 16 CE 67 4D 10 AE C1 10 :AF

```

Sum: E0 63 77 15 CC 23 B5 62 79 4A F1 13 F0 A9 EE 42 :35

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6E00 26 F7 4A 26 EF B6 39 A7 B4 BE 30 00 B6 B0 :BD
6E10 00 B6 1B 4A 26 FD 96 05 2B FC 1A 50 B6 :B0
6E20 C6 08 5A 27 FD 96 05 2A FC B6 01 B7 FC :B0
6E30 00 CE FC B1 C6 04 6F C4 64 B0 69 C4 5A :26
6E40 C4 68 C4 68 C4 68 C4 33 41 11 83 FD 00 :26
6E50 00 97 05 1C AF 20 00 BE CC 57 BF 01 E1 :7E
6E60 39 00 00 00 7F 6F 09 3B 4A 26 FD 96 05 :2B
6E70 50 B6 B0 97 05 C6 08 5A 27 FD 96 05 2A :FC
6E80 B7 FC B0 86 00 97 05 1C AF 35 12 3B 34 :7F
6E90 17 BE 6E 64 BF 01 E1 B6 B0 B7 6F 09 B6 :FD
6EA0 B6 7F BE 34 00 C6 12 CE 67 C9 10 AE C1 :10
6EB0 5A 26 F7 4A 26 EF B6 39 A7 B4 BE 30 00 :B6
6EC0 34 00 B6 1B 4A 26 FD 96 05 2B FC 1A 50 :B6
6ED0 05 C6 08 5A 27 FD 96 05 2A FC B6 01 B7 :FC
6EE0 30 00 CE FC B1 C6 03 6F C4 64 B0 69 C4 :5A
6EF0 6B C4 68 C4 68 C4 68 C4 33 41 11 83 FD :00

```

Sum: B8 91 3B CD 0E D4 94 67 55 A3 DD 4B C9 E2 DD 6F :42

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
6F00 26 E3 B6 00 97 05 1C AF 20 00 BE CC 57 :BF
6F10 7E 61 B8 39 7F 39 00 00 00 7F 6C 32 3B :86
6F20 26 FD 96 05 2B FC 1A 50 B6 B0 97 05 C6 :08
6F30 FD 96 05 2A FC B6 02 B7 FC B0 86 00 97 :05
6F40 35 7F 39 34 7F BD 60 17 BE 6F 19 BF 01 :E1
6F50 B7 6F C2 B6 FD 1F B8 B6 7F BE 34 00 C6 :0E
6F60 3C 10 AE C1 10 AF B1 5A 26 F7 4A 26 EF :B6
6F70 B4 BE 30 00 B6 B0 BD 34 00 B6 1B 4A 26 :FD
6F80 2B FC 1A 50 B6 B0 97 05 C6 08 5A 27 FD :96
6F90 FC B6 01 B7 FC B0 8E 30 00 CE FC B1 C6 :02
6FA0 64 B0 69 C4 5A 26 F9 33 41 11 83 FD 00 :26
6FB0 C4 68 C4 33 41 11 83 FC FF 2E E1 B6 00 :97
6FC0 AF 20 00 BE CC 57 BF 01 E1 7E 61 B8 39 :00

```

Sum: 71 ED FA 6F 3B 59 C1 7B 3F DB 25 10 B5 5B 56 35 :54



# 実戦 株式チャート

～株価の動きを適確に把握する～

## ■アイ&ケイ

以前、PC-8001で株式チャート・プログラムを作ろうとしたのですが、ドットが粗いので、データのロードに時間がかかり、あきらめていました。今度FM-8を購入してみて、ドットが細かく、1,600ボーンなので少しは速いだろうと思い、再度挑戦してみました。結果は予想以上でした。

チャートはかなり細くなり、それ以外にいろいろな機能を加えることができました。



## 機器構成

- ① FM-7/8
- ② 高解度ディスプレイ
- ③ フリント
- ④ カセット・レコーダ

プリントはなくても使えますが、当然ハード・コピーはできません。

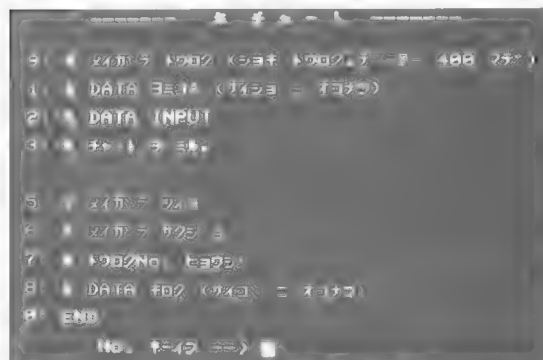


## プログラムについて

プログラムを作るときの基本的な考え方として、以下の点に注意しました。

- ① 入力するデータは週末の終値だけとする。  
高値、安値、出来高などの入力も考えましたが、入力に時間がかかって面倒であり、また余り多くの情報を入れても、かえって判断が鈍くなってしまうので、単純にしました。
- ② チャートのスケーリングはできるだけ同じとする。市販のチャート誌は銘柄によりまちまちなので、相対的に高

写真 1



いのか安いのか判断しにくいからです。

- ③ 売り買いのタイミングは、ハッチ法によるものとします。  
ハッチ法とは、安値より10%上昇したところで買い、高値より10%下がったところで売る方法です。

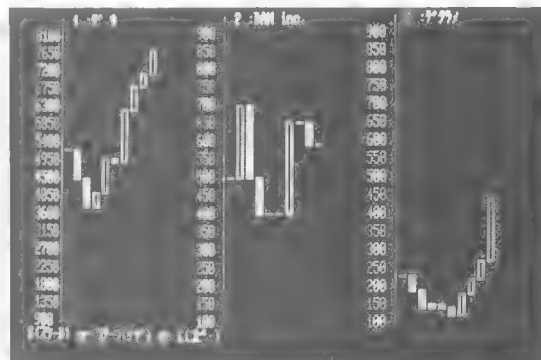


## 使い方

- ① キー入力について：基本的に、各作業の終了は「O」で終わるようにしてあります。また、途中に「Y・リ/N」とか「Y/N・リ」の表示ができますが、これは「Y」が「RETURN」のどちらでも受けつけるという意味です。後者の場合、「N」が「RETURN」です。多く使う方を、「RETURN」で処理できるようにしました。
- ② 登録：最初は「9」を押して初期登録から行なってください。その後は、「5」を押し、追加を行なってください。その場合、続けてデータを45週分まで入れることができます。「O」を入れると入力は終了し、高値のあった月を聞いてきます。それが終わると次の入力に移ります。なお、データを、あらかじめ読み込んであれば、間違っても「9」を押しても登録を受けつけないようにしてあります。
- ③ データ入力：1画面15銘柄ずつ入力、表示していきます。入力ミスを防ぐため、右側に先週終値を表示してあります。これによって異常値を見つけます。データが入りきらなくなると、「アフレ」と表示して次の銘柄に移ります。「RETURN」を押すと、各銘柄ごとに安値、高値の月を比較して入れ換えてゆき、次の画面に移ります。  
登録できる銘柄数は60銘柄までです。これ以上はデータ用テープを別に作ってください。

記録できるデータ数は2桁銘柄の場合、63週分（1

写真 2



年2箇月位), 4桁銘柄の場合、50週分(約1年分)です。これ以外の株価、つまり2桁とか5桁の場合は3, 4桁に修正して入力してください。

データのロード時間は、13銘柄10箇月分(43週分)を読み込んで約1分位です。60銘柄入れても4~5分で済むと思います。この程度であれば、イライラして待つこともないでしょう(プログラム自体は、約2分弱でロードできます)。

④銘柄を買った場合: まず **[7]** を押して銘柄の登録番号を調べてからメニューに戻り、**[4]** を押してください。ここで買値を入力すると、それ以降の株価が買値を上まわれば、自動的に高値と買い値の入れ換えを行ないます。これは、ハッチ法により売りのタイミングを見るときに必要なので設けました。

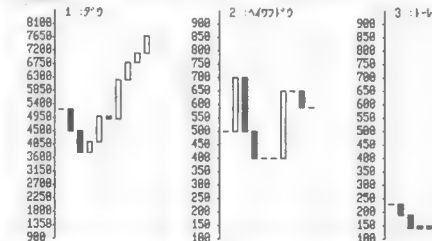
⑤チャート表示: ここに一番力を入れました。

まずチャート・メニューの中で、“UPメイガラ”とあるのは、今週と先週を比較して、上昇しているものを表示していきます。“+10%メイガラ”は、安値より10%アップした場合、“-10%メイガラ”は、高値(安値)より10%ダウンした場合のみ表示します。また、新高値銘柄は、新高値の場合のみ表示します。

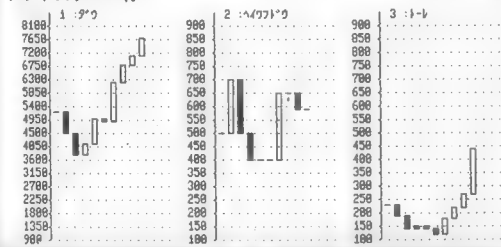
これらはすべて、今週のデータでのみ判断しています。つまり、先週のデータが+10%を越えていて、今週のデータも+10%を越えていた場合には表示しません。これ

年足チャート例

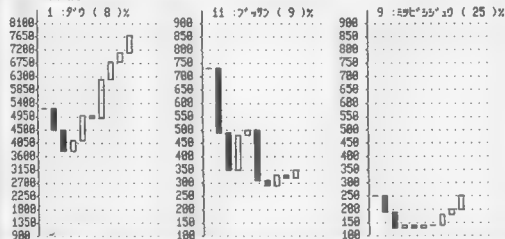
初期表示



ドットスケール付

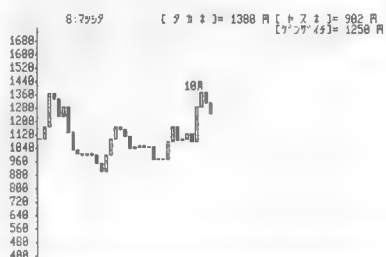


UP銘柄



週足チャート例

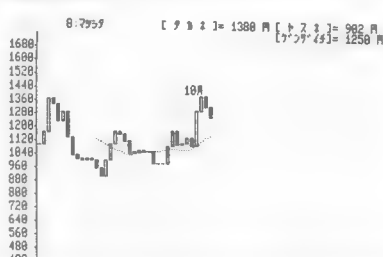
初期表示



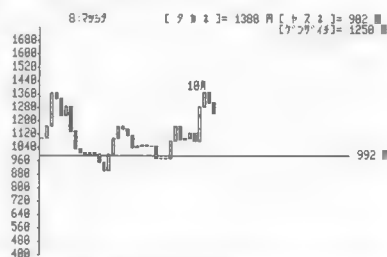
ドットスケール入り



13週移動平均線入り



+10%(カ)ライン入り



-10%(ウ)ライン入り損益表示





により最新情報のみを取り出せるわけです。

つぎにチャートそのものですが、**[1]~[6]**のキーについては、続けて押すと前の表示が消されます。**[1]**は横のライン引き、**[2]**は13週移動平均線（ただし、データが13週以上必要）でこれも売買の日安となります。**[3]**と**[4]**は、 $\pm 10\%$ 、 $-10\%$ のラインを引き、その値も表示します。**[5]**は、スケーリングを2倍にして（チャートから見れば1/2縮小）表示します。大きな流れを見たいときに使います。**[6]**は現在値に対する損益を表示します。株数は千株単位で入れてください。**[0]**で次に移ります。

**⑥銘柄削除**：この場合も登録番号を前もって見ておく必要があります。また、削除を行なうと、データ記録のとき、登録番号がズレて記録されるので、次にデータを読み込むとき、**[7]**を押して新しい番号を見ておいてください。

**⑦データ記録**：11日の作業の終わりには必ず行なってください。



## 年間チャートについて

### 週足プログラム

```

10 ***** シュウソク フロク ラム *****
20 '
11,7 By K.I
30 CLEAR17000:DEFINT A-Z
40 M1=60:M2=45
50 DIM N$(M1),D$(M1),G$(M1),DA$(M1),T$(M2),M(M1),
V(65),HI(M1),LO(M1),KA(M1)
60 WIDTH40,25:LOCATE0,0:COLOR7:PRINT"----- シ
ユウソク -----"
70 LOCATE 0,2:COLOR3:PRINT"9: メイカ ラ トロワ (ショキ トロワ
ク。データー":M1;"マテ。)"
80 LOCATE 0,4:COLOR6:PRINT"1: DATA ヨミコミ (サイショ ニ オ
ゴナウ)"
90 LOCATE 0,6:COLOR7:PRINT"2: DATA INPUT"
100 LOCATE 0,8:COLOR5:PRINT"3: チャート ラ ミル"
110 LOCATE 0,10:COLOR2:PRINT"4: メイカ ラ (カイ)"
120 LOCATE 0,12:COLOR4:PRINT"5: メイカ ラ ツイカ"
130 LOCATE 0,14:COLOR4:PRINT"6: メイカ ラ ウツシヨ"
140 LOCATE 0,16:COLOR5:PRINT"7: トロワ No. ヒョウシ"
150 LOCATE 0,18:COLOR6:PRINT"8: DATA キロク (サイコ) ニ
オコナウ"
160 LOCATE 0,20:COLOR7:PRINT"0: END":COLOR7
170 LOCATE 0,23:COLOR4:INPUT"No. キーイン --> ",Q
180 IF Q>9 THEN 170
190 IF Q=9 THEN 1380
200 IF Q=0 THEN WIDTH80,20:COLOR7:END
210 ON Q GO TO 1420,1560,1930,2380,2490,2780,289
0,2960
220 **** SUB ルーチン ****
230 ナシ
240 LOCATE25,10:PRINT"ナシ !!!":FOR M=1TO2000:NEXT
:RETURN1940
250 マエ ノ DATA
260 FOR R=1 TO E2
270 LE=LEN(D$(R))
280 DA$(R)=MID$(D$(R),LE-5,4)
290 IF INSTR(DA$(R),"+" )=0 THEN A=10 ELSE A=9
300 M(R)=VAL(MID$(D$(R),LE-A,4))
310 NEXT R:RETURN
320 カンリョウ
330 LOCATE15,15:COLOR4:PRINT"カンリョウ !!!":COLOR7:F
OR I=1 TO 1000:NEXT:RETURN60
340 END
350 CLS:LOCATE25,10:COLOR6:PRINT"E N D":FOR I=1T
O1000:NEXT:RETURN1940
360 トロワ
370 IF R>M1 THEN E2=R:LOCATE5,10:COLOR2:PRINT"メイ
カ ラ オーナ"-->:FOR I=1 TO 3000:NEXT:RETURN60
380 LOCATE0,2:COLOR4:PRINT"ケンサイ ノ トロワ No.":R
390 LOCATE17,6:PRINT"-----"
400 LOCATE0,5:COLOR7:INPUT"メイカ ラ (9ミシ イナイ)",N$(
R)
410 IF LEN(N$(R))>9 THEN BEEP:GO TO 400
420 HI(R)=100:KA(R)=100:LO(R)=20000:G$(R)="1":D$(
R)=" "
```

年間チャートも作成しておきました。過去10年間の年足を記録しておくためです。データ数は400まで。操作方法は週足チャートとほぼ同じです。この中で、**[3]**を押して、チャートを指定してアップ銘柄を選別させると、アップ率を表示して率の小さい順に表示していきます。このハード・コピーをとって新年度の作戦を立てるわけです。アップ率が小さいということは、新年度でさらに上昇する可能性が大きいと言えます。安値で買って高値で売るのは、株の基本ですから……。



## 終わりに

以上ですが、単にマイコンにチャートを描かせるのではなく、実際の売買に役立てるために使うことが大切です。このプログラム完成後、早速ある銘柄を買っていますが、現在、2週間でFM-8を買うだけの利益がでています。すべり出しは順調のようです。株に興味のある方は、打ち込んでみてください。

```

430 LOCATE5,19:COLOR6:INPUT"カクエン... OK ? (Y/N)"
",Q$:COLOR7
440 IF Q$="Y" OR Q$="" THEN 460
450 IF Q$="N" THEN CLS:GO TO 380 ELSE 430
460 LOCATE5,23:COLOR7:INPUT"ツキ カ アリマスカ ? (Y/N)"
",Q$
470 IF Q$="Y" OR Q$="" THEN CLS:R=R+1:GO TO 370
480 IF Q$="N" THEN E2=R:RETURN ELSE 460
490 ミタシ
500 LOCATE0,0:COLOR7:PRINT"-----"
"
510 LOCATE0,1:COLOR6:PRINT"No : メイカ ラ N
o : メイカ ラ No : メイカ ラ"
520 LOCATE0,2:COLOR7:PRINT"-----"
"
":RETURN
530 リ.C
540 LOCATE5,23:COLOR5:PRINT"リターン (ツキ) or C (コピー) "
:COLOR7
550 Q$=INKEY$
560 IF Q$="C" THEN LOCATE5,23:PRINTSPC(22):HARDC
:CLS:RETURN
570 IF Q$=CHR$(13) THEN CLS:RETURN ELSE 550
580 Y
590 IF HI(R)/CY <= (156+100/CS) THEN RETURN
600 CY=CY+CS:GO TO 590
610 クラフ
620 COLOR5:LOY=CY*(100/CS):S=LOY:SS=10*CY
630 LINE(43,10)-(43,180),PSET,4
640 FOR I=180 TO 20 STEP-10:S$=STR$(S)
650 IF (I MOD 20)=0 THEN 660 ELSE 670
660 COLOR6:SYMBOL(0,I-4),S$,1,1:GO TO 680
670 COLOR4:SYMBOL(0,I-4),S$,1,1
680 S=S+SS:NEXT
690 FOR I=180 TO 20 STEP -10:PSET(41,I):PSET(42,
I):NEXT
700 チャート
710 NE=0:IF LEFT$(D$(R),1)="" THEN NE=1:RETURN
720 ED=INSTR(D$(R),"")
730 V$=LEFT$(D$(R),INSTR(D$(R),"")-1):V=VAL(V$)
740 YO=180-(V-LOY)/CY:IF YO>180 THEN YO=180
750 T=1:NV=0:FS=0:CC=1:H6$=G$(R)+"月"
760 IF T=ED THEN TC=CC:LOCATE25,0:COLOR6:PRINT"[
タ カ ネ ]=":HI(R);"月":LOCATE43,0:COLOR5:PRINT"[
タ ス ネ ]=":LO(R);"日":LOCATE43,1:COLOR7:PRINT"[ケンサ
イ]":NV;"日":RETURN
770 V$=MID$(D$(R),T,INSTR(T,D$(R),"")-T)
780 V=VAL(V$):V(CC)=V:L=LEN(V$):T=T+L+1
790 Y=180-(V-LOY)/CY:IF Y>180 THEN Y=180ELSE IF
Y<10 THEN Y=10
800 IF Y>YO THEN 810 ELSE 820
810 LINE(X0,Y0)-(X0+3,Y),PSET,5,BF:GO TO 840
820 LINE(X0,Y0)-(X0+3,Y),PSET,7,B
```

## 週足プログラム

```

830 IF V=HI(R) AND FS=0 THEN FS=1:SYMBOL(X0-20,Y
-B),HB#,1,1,5
840 Y0=Y:X0=X0+6:NV=V:CC=CC+1:GO TO 760
850 '13 シュウ
860 IF V(13)=0 THEN BEEP:LOCATE63,17:COLOR2:PRIN
T"デ-9- フロク!!":FOR J=1 TO 1000:NEXT:LOCATE63,
17:PRINTSPC(15):RETURN
870 CC=CC-13:TX=118:T1=0
880 FOR J=1 TO 13:T1#=(T1#+(V(J)):NEXT:T2=CINT(T1#
/13)
890 TY=180-(T2-LOY)/CY:IF TY>180 THEN TY=180 ELS
E IF TY<10 THEN TY=10
900 PSET(TX,TY,6,XOR)
910 T1#=0:TX=TX+6:DTY=TY
920 FOR I=2 TO CC
930 FOR J=1 TO I+12:T1#=(T1#+(V(J)):NEXT:T2=CINT(T1
#/13)
940 TY=180-(T2-LOY)/CY:IF TY>180 THEN TY=180 ELS
E IF TY<10 THEN TY=10
950 DTY=(DTY+TY)/2
960 PSET(TX-3,DTY,6,XOR)
970 PSET(TX,TY,6,XOR)
980 T1#=0:TX=TX+6:DTY=TY:NEXT
990 RETURN
1000 'DOT
1010 FOR I=180 TO 20 STEP-10:FOR J=45 TO 432 STE
P 10:PSET(J,1,4,XOR):NEXTJ,I:RETURN
1020 '10%
1030 VD=INT(KA(R)*.9):VD$=STR$(VD)+" "R":YD=180-(
VD-LOY)/CY:IF YD<10 THEN YD=10 ELSE IF YD>180 TH
EN YD=180
1040 LINE(45,YD)-(432,YD),XOR,2:SYMBOL(435,YD-4)
,VD$,1,1,7,XOR:RETURN
1050 '10%
1060 VD=INT(LO(R)*1.1):VD$=STR$(VD)+" "R":YD=180-(
VD-LOY)/CY:IF YD<10 THEN YD=10 ELSE IF YD>180 TH
EN YD=180
1070 LINE(45,YD)-(432,YD),XOR,5:SYMBOL(435,YD-4)
,VD$,1,1,7,XOR:RETURN
1080 'ソシキ
1090 LOCATE60,19:COLOR5:INPUT"カ イ ネ =" ,A1
1100 LOCATE60,21:COLOR6:PRINT"カフスク =" :LOCATE72,
21:COLOR6:PRINT"(*1000)"
1110 LOCATE67,21:COLOR6:INPUT" ",A2:AT=CINT(((NV
-A1*1.03)*A2)/10)
1120 IF AT<0 THEN 1140 ELSE 1130
1130 LOCATE60,23:COLOR4:PRINT"ソシキ= ア7 +":AT;"マ
ン"
1140 LOCATE60,23:COLOR2:PRINT"ソシキ= ア7 " :AT;"マ
ン"
1150 'ヒョウ
1160 WIDTH80,25:X0=45:LOCATE 8,0:COLOR7:PRINTUSI
NG"####&
&";R,N$(R)
1170 GOSUB580:GOSUB610:"レンシ"&,"ラフ
1180 LOCATE63,1:COLOR4:PRINT"1: ドット スケール"
1190 LOCATE63,3:COLOR6:PRINT"2: 13シュウ イトウイキチン"
1200 LOCATE63,5:COLOR2:PRINT"3: -10%(ウリ) ライン "
1210 LOCATE63,7:COLOR5:PRINT"4: +10%(カイ) ライン "
1220 LOCATE63,9:COLOR5:PRINT"5: (1/2) シュクショウ"
1230 LOCATE63,11:COLOR3:PRINT"6: ソシキ"
1240 LOCATE63,13:COLOR4:PRINT"7: コ ヒ-ー"
1250 LOCATE63,15:COLOR7:PRINT"0: ツ キ-ウツル."
1260 LINE(495,0)-(639,128),PSET,4,B
1270 LOCATE63,17:COLOR6:PRINT"No.ラ ケ-イン --->"
1280 Q$=INKEY$:IF Q$="" THEN 1280
1290 LOCATE63,17:PRINTSPC(15)
1300 IF Q$="0" THEN CLS:V(13)=0:X=45:YN=80:RETUR
N
1310 IF Q$="1" THEN GOSUB1000:GO TO 1270
1320 IF Q$="2" THEN CC=TC:GOSUB850:GO TO 1270
1330 IF Q$="3" THEN GOSUB1020:GO TO 1270
1340 IF Q$="4" THEN GOSUB1050:GO TO 1270
1350 IF Q$="5" THEN IF F5=0 THEN F5=1:CY=2*CY:CS
=CY:GO TO 1150 ELSE F5=0:CY=CY/2:CS=CY:GO TO 1150
1360 IF Q$="6" THEN IF F6=0 THEN F6=1:GOSUB1080:
GO TO 1270 ELSE F6=0:LINE(470,150)-(639,199),PSE
T,0,BF:GO TO 1270
1370 IF Q$="7" THEN LINE(495,0)-(639,136),PSET,0
,BF:HARDC2:GO TO 1180 ELSE BEEP:GO TO 1270
1380 ' トロク(シュキ)
1390 WIDTH40,25:IF HI(1)<>0 THEN BEEP:LOCATE0,10
:PRINT"トロク アリ!! (5: ツイカ) ラ シテワザイ":FOR I=1 TO
3000:NEXT:GO TO 60
1400 R=1:GOSUB360:"トロク
1410 GO TO 60
1420 ' ヨミコミ

```

```

1430 WIDTH40,20:LOCATE5,10:COLOR5:PRINT"DATA テ-7
" :COLOR7:PRINT"ラ セット シテ"
1440 LOCATE10,12:COLOR6:PRINT" PLAY ":COLOR7:PR
INT"ニ シテ ワザザイ."
1450 LOCATE5,15:PRINT"( オウイ) ":COLOR5:PRINT"RE
TURN ケ-":COLOR7:PRINT"ラ ス."
1460 IF INKEY$=CHR$(13) THEN 1470 ELSE 1460
1470 CLS:LOCATE5,10:PRINT"--- ヨミコミ シュウ ---"
1480 R=1:OPEN"1",#2,"CASO:SYUDATA1"
1490 IF EOF(2) THEN CLOSE#2:GO TO 1550
1500 INPUT #2,HI(R),LO(R),KA(R)
1510 INPUT #2,N$(R)
1520 INPUT #2,G$(R)
1530 INPUT #2,D$(R)
1540 R=R+1:GO TO 1490
1550 E2=R-1:GOSUB320
1560 ' DATA INPUT
1570 WIDTH40,20:LOCATE5,10:COLOR7:INPUT"ナン 月 テス
カ =" ,G$:CLS
1580 X=0:XX=24:FLAG=1
1590 'ヒョウシ & INPUT
1600 FOR R=1 TO E2 STEP 15
1610 LOCATE0,0:COLOR6:PRINT" No: メイカ ラ : コンシ
ウ ":COLOR4:PRINT" ショウ"
1620 LOCATE0,1:PRINT"
"
1630 LOCATE0,17:PRINTSTRING$(31,"-")
1640 Y=2:FOR J=R TO R+14
1650 LE=LEN(D$(J)):IF LE=2 THEN DD=0:GO TO 1680
1660 IF LE>250 THEN BEEP:LOCATE0,18:COLOR2:PRINT
N$(J):" = ア7レ":DA$(J)="":DA=0:FOR M=1 TO2000:NEXT
DD=VAL(MID$(D$(J),LE-5,4)):M(J)=DD:LOCATE0,18:P
RINTSPC(20):GO TO 1690
1670 DD=VAL(MID$(D$(J),LE-5,4)):M(J)=DD
1680 COLOR7:LOCATE0,18:PRINTN$(J):"=" :INPUT" ",
DA$(J):DA=VAL(DA$(J)):LOCATE0,18:PRINTSPC(20)
1690 LOCATE0,Y:COLOR7:PRINTUSING"####&
&":J,N$(J),DA
1700 LOCATE23,Y:COLOR4:PRINTUSING"#####":DD
1710 Y=Y+1:IF J>E2-1 THEN BEEP:GO TO 1730
1720 NEXT J:BEEP
1730 LOCATE0,18:COLOR6:INPUT"Z (タイセイ), リターン(ツキ) "
,Q$:COLOR7:LOCATE0,18:PRINTSPC(30)
1740 IF Q$="Z" THEN 1760
1750 IF Q$="" THEN 1810ELSE 1730
1760 'タイセイ
1770 LOCATE5,18:INPUT"タイセイ スル No=" ,Q:LOCATE5,18
:PRINTSPC(20)
1780 IF Q<R OR Q>R+14 OR Q>E2 THEN BEEP:GO TO 17
70
1790 IF VAL(DA$(Q))=0 THEN BEEP:GO TO 1730
1800 LOCATE0,18:COLOR5:PRINT"イマノ テ-ター=" :DA$(Q):
LOCATE20,18:COLOR7:INPUT"タダシイ テ-ター=" ,DA$(Q):DA
=VAL(DA$(Q)):LOCATE16,Q-(Q#16)*15+1:COLOR6:PRINT
USING"#####":DA:COLOR7:LOCATE0,18:PRINTSPC(38):G
O TO 1730
1810 ' +
1820 FOR J=R TO R+14
1830 DA=VAL(DA$(J))
1840 IF DA=0 THEN 1890
1850 IF DA>HI(J) THEN HI(J)=DA:G$(J)=G$
1860 IF DA<LO(J) THEN LO(J)=DA
1870 IF DA>KA(J) THEN KA(J)=DA
1880 D$(J)=LEFT$(D$(J),INSTR(D$(J),"=")-1)+DA$(J
)+"+"
1890 IF J>E2-1 THEN 1920
1900 NEXT J
1910 CLS:NEXT R
1920 GO TO 60
1930 ' ショウ ナート
1940 WIDTH40,25:LOCATE3,3:COLOR7:PRINT"----- ナ
ート -----"
1950 LOCATE3,6:COLOR7:PRINT"1: センフ"
1960 LOCATE3,8:COLOR7:PRINT"2: コ ヒ-ー"
1970 LOCATE3,10:COLOR5:PRINT"3: UP メイカ"ラ"
1980 LOCATE3,12:COLOR4:PRINT"4: シン クカネ メイカ"ラ"
1990 LOCATE3,14:COLOR5:PRINT"5: +10%(カイ) メイカ"ラ"
2000 LOCATE3,16:COLOR2:PRINT"6: -10%(ウリ) メイカ"ラ"
2010 LOCATE3,18:COLOR7:PRINT"0: ニュー ニ モトル"
2020 LOCATE3,23:INPUT"No. ケ-イン ---> " ,A
2030 IF A>6 THEN BEEP:GO TO 2020
2040 IF A=0 THEN 60
2050 ON A GO TO 2060,2100,2180,2230,2280,2330
2060 ' センフ
2070 WIDTH80,25:FOR R=1 TO E2:CY=2:CS=CY
2080 GOSUB1150:"ア7ラフ

```

```

2090 NEXT R:GOSUB340:END
2100 'コハツ
2110 WIDTH40,25:CY=2:CS=CY
2120 LOCATE0,0:COLOR4:PRINT"クレーン スウ=";E2;"
"
2130 LOCATE0,3:COLOR7:INPUT"トウロ? No. (0:END) =";R
2140 IF R=0 THEN 1940
2150 IF R>E2 THEN BEEP:GOTO 2130
2160 GOSUB1150:'クレーン
2170 GO TO 2110
2180 'UP
2190 IF FLAG=0 THEN GOSUB250:FLAG=1
2200 WIDTH80,25:CA=0:FOR R=1 TO E2:CY=2:CS=CY
2210 IF M(R)<VAL(DA$(R)) THEN CA=1:GOSUB1150:'クレーン
2220 NEXT R:IF CA=0 THEN GOSUB230 ELSE GOSUB340:'END
2230 'シタカネ
2240 IF FLAG=0 THEN GOSUB250:FLAG=1
2250 WIDTH80,25:CA=0:FOR R=1 TO E2:CY=2:CS=CY
2260 IF VAL(DA$(R))>HI(R) THEN CA=1:GOSUB1150
2270 NEXT R:IF CA=0 THEN GOSUB230 ELSE GOSUB340
2280 '10%
2290 IF FLAG=0 THEN GOSUB250:FLAG=1
2300 WIDTH80,25:CA=0:FOR R=1 TO E2:CY=2:CS=CY
2310 IF VAL(DA$(R))>LO(R)*1.1 AND M(R)<LO(R)*1.1 THEN CA=1:GOSUB1150
2320 NEXT R:IF CA=0 THEN GOSUB230 ELSE GOSUB340
2330 '10%
2340 IF FLAG=0 THEN GOSUB250:FLAG=1
2350 WIDTH80,25:CA=0:FOR R=1 TO E2:CY=2:CS=CY
2360 IF VAL(DA$(R))<KA(R)*.9 AND M(R)>KA(R)*.9 THEN CA=1:GOSUB1150
2370 NEXT R:IF CA=0 THEN GOSUB230 ELSE GOSUB340
2380 'カイ
2390 WIDTH40,20:LOCATE5,5:INPUT"トウロ? No=";R
2400 IF R>E2 THEN BEEP:GO TO 2390
2410 LOCATE10,7:COLOR6:PRINT N$(R)
2420 LOCATE5,9:COLOR6:INPUT"カネ? OK? (Y/N)"
2430 IF Q$="Y" OR Q$="" THEN 2450
2440 IF Q$="N" THEN 2390 ELSE 2420
2450 LOCATE5,10:INPUT"カネ = ";K:KA(R)=K
2460 LOCATE5,15:COLOR6:INPUT"ツキ? カ アリマスカ? (Y/N)"
2470 IF Q$="Y" THEN 2390
2480 IF Q$="N" OR Q$="" THEN 60 ELSE 2460
2490 'ツキ
2500 WIDTH40,25:E2=E2+1:R=E2:DE2=E2
2510 GOSUB360:'トウロ
2520 WIDTH80,20:FOR R=DE2 TO E2
2530 LOCATE0,0:PRINT R;" ";N$(R):LOCATE20,0:PRINT" ( DATA ノ カス = ";M2;"マテ。 0: END )
2540 'INPUT
2550 X=0:Y=2:C=1
2560 LOCATEX,Y:PRINTUSING"###=";C;:INPUT" ",T$(C)
2570 IF T$(C)="0" THEN E1=C-1:GO TO 2600
2580 Y=Y+1:IF Y>16 THEN Y=2:X=X+20
2590 C=C+1:IF C>M2 THEN E1=C-1:GO TO 2600 ELSE 2560
2600 BEEP:LOCATE 5,18:COLOR6:INPUT"Z(タイセイ) リターン(ツキ)"
2610 IF Q$="Z" THEN 2630
2620 IF Q$="" THEN 2670 ELSE 2600
2630 'タイセイ
2640 LOCATE5,18:INPUT"タイセイ スル No=";Q:LOCATE5,18:PRINTSPC(20)
2650 IF Q>E1 THEN BEEP:GO TO 2640

```

```

2660 LOCATE0,18:COLOR5:PRINT"イマノ テーラー=";T$(Q):LOCATE20,18:COLOR7:INPUT"タタシイ テーラー=";T$(Q):XA=Q*16:LOCATE XA*20+5,Q-XA*15+1:COLOR6:PRINTT$(Q);"
";COLOR7:LOCATE0,18:PRINTSPC(38):GO TO 2600
2670 'LO,HI
2680 FOR J=1 TO E1:T=VAL(T$(J))
2690 IF T>HI(R) THEN HI(R)=T
2700 IF T<KA(R) THEN KA(R)=T
2710 IF T<LO(R) THEN LO(R)=T
2720 NEXT J
2730 BEEP:LOCATE0,18:PRINT"タカネ =";HI(R);" 円":LOCATE20,18:INPUT"タカネ ノ ナンバ デスカ =";G$(R):LOCATE0,18:PRINTSPC(45)
2740 '10%
2750 DA$="":FOR J=1 TO E1:DA$=DA$+T$(J)+" ";NEXT J
2760 D$(R)=DA$+"="
2770 CLS:NEXT R:GO TO 60
2780 'サワシヨ
2790 WIDTH40,20:LOCATE5,3:INPUT"サワシヨ スル トウロ? No. = ";R
2800 IF R>E2 THEN BEEP:GO TO 2790
2810 LOCATE10,7:COLOR5:PRINT N$(R)
2820 LOCATE5,10:COLOR7:INPUT"サワシヨ シテ OK? (Y/N)"
2830 IF Q$="Y" OR Q$="" THEN 2850
2840 IF Q$="N" THEN CLS:GO TO 2790 ELSE 2820
2850 N$(R)="
2860 LOCATE5,15:COLOR6:INPUT"ツキ? カ アリマスカ? (Y/N)"
2870 IF Q$="Y" THEN CLS:GO TO 2790
2880 IF Q$="N" OR Q$="" THEN 60 ELSE 2860
2890 'トウロ? No. ヒョウシ
2900 WIDTH80,25:GOSUB490:'ミタシ
2910 X=0:Y=3:COLOR7
2920 FOR R=1 TO E2
2930 LOCATEX,Y:PRINTUSING"###: & &";R,N$(R)
2940 Y=Y+1:IF Y>17 THEN Y=3:X=X+20:IF X>60 THEN GOSUB530:GOSUB 490:X=0:'リ,C
2950 NEXT:GOSUB530:GO TO 60:'リ,C
2960 'DATA 407
2970 WIDTH40,20:LOCATE5,10:COLOR5:PRINT"DATA テーラー"
2980 LOCATE10,12:COLOR6:PRINT"ロクオン ";:COLOR7:PRINT"ニ シテ フタ サイ。"
2990 LOCATE5,15:PRINT"( オウイ )"
3000 IF INKEY$=CHR$(13) THEN 3010 ELSE 3000
3010 CLS:LOCATE5,10:COLOR5:PRINT"--- 407 シュウ ---"
3020 OPEN"D",#2,"CAS0:SYUDATA1"
3030 FOR R=1 TO E2
3040 IF N$(R)="" THEN 3090
3050 PRINT#2,HI(R):LO(R):KA(R)
3060 PRINT#2,N$(R)
3070 PRINT#2,G$(R)
3080 PRINT#2,D$(R)
3090 NEXT:CLOSE#2:GOSUB320

```

## 年足プログラム

```

10 ***** NEN フロム *****
20
1,7 By K.I
30 CLEAR4000:DEFINT A-Z
40 M2=400
50 DIM Z(M2,10),HI(M2),LO(M2),M(M2,1),ST(M2,1),N$(M2)
60 WIDTH40,25:LOCATE5,0:COLOR7:PRINT"----- 年
チー ト
70 LOCATE0,3:COLOR3:PRINT"9: 年 メイカラ トウロ? (ショキ
トウロ.テーラー:M2:"マテ")
80 LOCATE0,5:COLOR6:PRINT"1: 年 DATA ヨスキ (サイシ
オコウ)"
90 LOCATE0,7:COLOR7:PRINT"2: 年 DATA INPUT"

```

```

100 LOCATE0,9:COLOR5:PRINT"3: 年 チー ト ラ ミル。"
110 LOCATE0,11:COLOR2:PRINT"4: フロム ミ DATA ンコウ"
120 LOCATE0,13:COLOR4:PRINT"5: 年 メイカラ ツイカ"
130 LOCATE0,15:COLOR4:PRINT"6: 年 メイカラ サワシヨ"
140 LOCATE0,17:COLOR5:PRINT"7: 年 トウロ?No. ヒョウシ"
150 LOCATE0,19:COLOR6:PRINT"8: 年 DATA 407 (サイコ
ニ オコウ)"
160 LOCATE0,21:COLOR7:PRINT"0: END":COLOR7
170 LOCATE5,23:INPUT" No. キーイン --> ",A
180 IF A>9 THEN 170
190 IF A=9 THEN 1110
200 IF A=0 THEN WIDTH80,20:END
210 ON A GO TO 1150,1280,1620,2170,2290,2330,2450,2550

```

## 年足プログラム

```

220 ' **** SUB ルーチン ****
230 ' --- トロフ ---
240 IF R>M2 THEN BEEP:LOCATE5,10:COLOR2:PRINT"メイ
カラ オーバー":FOR I=1 TO 3000:NEXT:RETURN60
250 LOCATE5,10:INPUT"ナン デ スカ = ",NE:CLS
260 LOCATE0,0:COLOR4:PRINT"ゲンゲイノ トロフ No.=":R
270 LOCATE17,5:PRINT"-----"
280 LOCATE0,4:COLOR7:INPUT"メイカラ (9モシイタイ) =",N$
(R)
290 IF LEN(N$(R))>9 THEN BEEP:GO TO 280
300 HI(R)=100:LO(R)=20000
310 FOR I=1 TO 10:Z(R,I)=0:NEXT
320 LOCATE5,18:COLOR6:INPUT"カクニ...OK? (Y・リ/N)
",Q$:COLOR7
330 IF Q$="Y" OR Q$="" THEN 350
340 IF Q$="N" THEN CLS:GO TO 260 ELSE 320
350 LOCATE5,23:COLOR7:INPUT"ツキ" カ アマスカ? (Y・リ/N)
",Q$
360 IF Q$="Y" OR Q$="" THEN R=R+1:CLS:GO TO 260
370 IF Q$="N" THEN E2=R:WIDTH40,20:GO TO 380 ELSE
E 350
380 FOR R=DE2 TO E2
390 LOCATE 0,0:PRINTUSING" ###:& &": R,N$(
R)
400 Y=3:FOR I=1 TO 10:LOCATE0,Y:PRINTUSING"## 年
=";NE+I-10
410 LOCATE 6,Y:INPUT" ",Z(R,I)
420 Y=Y+1:NEXT I
430 LOCATE5,18:COLOR6:INPUT"Z(タイセイ),リターン(ツキ) ",
Q$:LOCATE5,18:PRINTSPC(25):COLOR7
440 IF Q$="Z" THEN 460
450 IF Q$="" THEN 490 ELSE 430
460 'タイセイ
470 LOCATE5,18:INPUT"タイセイ スル 年 = ",B:B=B+10-NE:L
OCATE5,18:PRINTSPC(20)
480 LOCATE0,18:COLOR5:PRINT"イマノ テーター=":Z(R,B):L
OCATE20,18:COLOR7:INPUT"タダシイ テーター= ",Z(R,B):LO
CATE 6,B+2:COLOR6:PRINTZ(R,B):" ":COLOR7:LOCATE
0,18:PRINTSPC(38):GO TO 430
490 'HI,LO
500 FOR I=1 TO 10
510 IF Z(R,I)>HI(R) THEN HI(R)=Z(R,I)
520 IF Z(R,I)<LO(R) THEN LO(R)=Z(R,I)
530 NEXT I
540 CLS:NEXT R
550 RETURN
560 ' --- ミタシ ---
570 LOCATE0,0:COLOR7:PRINT"-----"
"
580 LOCATE0,1:COLOR6:PRINT"No : 年メイカラ No : 年
メイカラ No : 年メイカラ No : 年
メイカラ "
590 LOCATE0,2:COLOR7:PRINT"-----"
"
600 RETURN
610 ' --- リ.C ---
620 IF HD=0 THEN 630 ELSE 640
630 LOCATE0,23:COLOR5:PRINT" リターン(ツキ) or C(コヒ-
)":COLOR7:GO TO 650
640 LOCATE0,23:COLOR5:PRINT"S(スケール) or リターン(ツキ)
or C(コヒ-)":COLOR7
650 Q$=INKEY$
660 IF Q$="S" THEN LOCATE0,23:PRINTSPC(35):GOSUB
910:GO TO 620
670 IF Q$="C" THEN LOCATE0,23:PRINTSPC(35):IF HD
=2 THEN HARD2:CLS:RETURN ELSE HARD2:CLS:RETURN
680 IF Q$=CHR$(13) THEN CLS:RETURN ELSE 650
690 ' --- Y ---
700 IF HI(R)/CY<170+100/CS THEN RETURN
710 CY=CY+CS:GO TO 700
720 ' --- クラフ ---
730 COLORS:LOY=CY*(100/CS):S=LOY:SS=10*CY:XS=X-4
3
740 LINE(X,10)-(X,180),PSET,4
750 FOR I=180 TO 20 STEP-10
760 S$=STR$(S)
770 IF (I MOD 20)=0 THEN 780 ELSE 790
780 COLOR6:SYMBOL(XS,I-4),S$,1,1:GO TO 800
790 COLOR4:SYMBOL(XS,I-4),S$,1,1
800 S=S+SS:NEXT
810 FOR I=180 TO 20 STEP -10:PSET(X-2,I):PSET(X-
1,I):NEXT
820 'チャート
830 Y0=180-(Z(R,1)-LOY)/CY:IF Y0>180 THEN Y0=180
840 FOR I=1 TO 10
850 Y=180-(Z(R,I)-LOY)/CY:IF Y>180 THEN Y=180 EL
SE IF Y<24 THEN Y=24
860 IF Y>Y0 THEN 870 ELSE 880
870 LINE(X0,Y0)-(X0+6,Y),PSET,5,BF:GO TO 890
880 LINE(X0,Y0)-(X0+6,Y),PSET,7,B
890 Y0=Y:X0=X0+12:NEXT
900 RETURN
910 ' --- トロフ ---
920 XM=45:FOR M=1 TO CC:FOR M1=180 TO 20 STEP -1
0:FOR M2=XM TO XM+150 STEP 10
930 PSET(M2,M1,4,XOR)
940 NEXT M2,M1
950 XM=XM+208:NEXT M
960 RETURN
970 'ソート
980 SR=0
990 IF SH-SL<=0 THEN 1100
1000 SX=M(SH,1):SJ=SH:SI=SL-1
1010 SJ=SJ-1
1020 IF SX<M(SJ,1) AND SJ>=SL GO TO 1010
1030 SI=SI+1
1040 IF SX>M(SI,1) GO TO 1030
1050 IF SI<SJ AND SJ>=SL THEN SWAP M(SI,0),M(SJ,
0):SWAP M(SI,1),M(SJ,1)
1060 IF SI<SJ GO TO 1010
1070 SWAP M(SI,0),M(SH,0):SWAP M(SI,1),M(SH,1)
1080 IF SI<SL<SH-SI THEN SR=SR+1:ST(SR,0)=SI+1:S
T(SR,1)=SH:SH=SI-1:GO TO 1090 ELSE SR=SR+1:ST(SR
,0)=SL:ST(SR,1)=SI-1:SL=SI+1
1090 GOSUB990:SL=ST(SR,0):SH=ST(SR,1):SR=SR-1:GO
SUB990
1100 RETURN
1110 ' 年トロフ
1120 WIDTH40,25:IF HI(1)<>0 THEN BEEP:LOCATE0,10
:PRINT"トロフアリ!!。(4: ツイ) ラ シテワサイ.":FOR I=1
TO 3000:NEXT:GO TO 60
1130 R=1:DE2=R:GOSUB230:'トロフ
1140 GO TO 60
1150 ' DATA 3535
1160 WIDTH40,20:LOCATE5,5:COLOR5:PRINT"DATA テーフ
":COLOR7:PRINT"ラ セット シテ"
1170 LOCATE10,7:COLOR6:PRINT"PLAY ":COLOR7:PRIN
T"ニ シテワサイ。"
1180 LOCATE5,12:PRINT"( オロイ) ":COLOR6:PRINT"RE
TURN 年":COLOR7:PRINT" ラ ス。)"
1190 IF INKEY$=CHR$(13) THEN 1200 ELSE 1190
1200 CLS:LOCATE5,10:COLOR5:PRINT"--- 3535 年 ---"
"
1210 R=1:OPEN"1",#2,"CASO:DATA"
1220 IF EOF(2) THEN CLOSE#2:GO TO1270
1230 INPUT #2,N$(R)
1240 INPUT #2,HI(R),LO(R)
1250 INPUT #2,Z(R,1),Z(R,2),Z(R,3),Z(R,4),Z(R,5)
,Z(R,6),Z(R,7),Z(R,8),Z(R,9),Z(R,10)
1260 R=R+1:GO TO 1220
1270 E2=R-1:LOCATE15,15:COLOR4:PRINT"カンヨウ !!":C
OLOR7:FOR I=1 TO 1000:NEXT:GO TO 60
1280 ' DATA INPUT
1290 WIDTH40,20
1300 FOR R=1 TO E2 STEP 15
1310 '1年 スラシ
1320 FOR J=R TO R+14
1330 Z(J,1)=0:FOR I=1 TO 9:Z(J,I)=Z(J,I+1):NEXT
1340 Z(J,10)=0
1350 IF J>E2-1 THEN 1370
1360 NEXT J
1370 'ヒョウシ & INPUT
1380 LOCATE0,0:COLOR7:PRINT" No: メイカラ : コト
シ":COLOR4:PRINT" : キョソソ"
1390 LOCATE0,1:COLOR4:PRINT"-----"
"
1400 LOCATE0,17:PRINTSTRING$(32,"-")
1410 Y=2:FOR J=R TO R+14
1420 COLOR7:LOCATE0,18:PRINTN$(J):"=":"INPUT" ",
Z(J,10):LOCATE0,18:PRINTSPC(20)
1430 LOCATE 0,Y:PRINTUSING"###: & &:#####
";J,N$(J),Z(J,10)
1440 LOCATE23,Y:COLOR4:PRINTUSING"#####:Z(J,9)
1450 IF J>E2-1 THEN BEEP:GO TO 1470
1460 Y=Y+1:NEXT J:BEEP
1470 LOCATE0,18:COLOR6:INPUT"Z(タイセイ),リターン(ツキ) "
,Q$:LOCATE0,18:PRINTSPC(25):COLOR7
1480 IF Q$="Z" THEN 1500
1490 IF Q$="" THEN 1540 ELSE 1470
1500 'タイセイ
1510 LOCATE5,18:INPUT"タイセイ スル No = ",Q:LOCATE5,18
:PRINTSPC(20)
1520 IF Q<R OR Q>R+14 OR Q>E2 THEN BEEP:GO TO 15
10
1530 LOCATE0,18:COLOR5:PRINT"イマノ テーター=":Z(Q,10)
:LOCATE20,18:COLOR7:INPUT"タダシイ テーター= ",Z(Q,10)
:LOCATE16,Q-(Q#16)*15+1:COLOR6:PRINTUSING"#####
":Z(Q,10):COLOR7:LOCATE0,18:PRINTSPC(38):GO TO 14

```

```

70
1540 'HI,LO
1550 FOR J=R TO R+14
1560 IF Z(J,I)>HI(J) THEN HI(J)=Z(J,I)
1570 IF Z(J,I)<LO(J) THEN LO(J)=Z(J,I)
1580 IF J>E2-1 THEN 1610
1590 NEXT J
1600 CLS:NEXT R
1610 GO TO 60
1620 ' --- 年チャート ---
1630 WIDTH40,20:CY=5:CS=CY
1640 LOCATES,1:PRINT"--- 年チャート ---"
1650 LOCATES,4:COLOR7:PRINT"1: センソフ"
1660 LOCATES,6:COLOR6:PRINT"2: コウツ"
1670 LOCATES,8:COLOR5:PRINT"3: UP メイカ"ラ"
1680 LOCATES,10:COLOR7:PRINT"0: メニュー ニ モト"ル。"
1690 LOCATES,15:INPUT"No. ラ キーイン ---> ",Q
1700 IF Q=1 THEN 1740
1710 IF Q=2 THEN 1850
1720 IF Q=3 THEN 1950
1730 IF Q=0 THEN 60ELSE 1690
1740 ' --- センソフ ---
1750 WIDTH80,25:X=43:YN=6:CC=0
1760 FOR R=1 TO E2:CY=5:CS=CY
1770 LOCATE XN,1:COLOR7:PRINT R;"":N$(R)
1780 GOSUB690:'レンシ"
1790 X=X+5:GOSUB720:'ワラフ
1800 X=X+20B:YN=XN+26:CC=CC+1:IF X>460 THEN HD=2
1800 GOSUB610:X=43:YN=6:CC=0:'リ,C
1810 NEXT R
1820 GOSUB610:'リ,C
1830 GO TO 1630
1840 ' --- コウツ ---
1850 WIDTH40,20:X=43:YN=6:CC=1:CY=5:CS=CY
1860 LOCATE0,0:COLOR4:PRINT"< テーター スウ = ":E2;"
>"
1870 LOCATES,10:COLOR7:INPUT"トワフ No. ( 0:END ) =
",R
1880 IF R=0 THEN 1630
1890 IF R>E2 THEN BEEP:GO TO 1870
1900 WIDTH80,25:LOCATE XN,1:COLOR7:PRINT R;"":N
$(R)
1910 GOSUB690:'レンシ"
1920 X=X+5:GOSUB720:'ワラフ
1930 HD=2:GOSUB610:'リ,C
1940 GO TO 1850
1950 ' --- UP メイカ"ラ ---
1960 CLS:C=1
1970 FOR R=1 TO E2
1980 ZD=Z(R,10)-Z(R,9)
1990 IF ZD>0 THEN M(C,0)=R:M(C,1)=(ZD/Z(R,9))*10
0:C=C+1
2000 NEXT R
2010 WIDTH40,25:LOCATES,10:COLOR5:PRINT"< テーター
スウ =":C-1;" >"
2020 LOCATES,13:PRINT"UP リツ ノ タイサイ シ"ユニ ニ ナラ"カエ
キウ"
2030 'ソート
2040 SL=1:SH=C-1:GOSUB970:'ソート
2050 'ヒョウシ"
2060 WIDTH80,25
2070 CLS:X=43:YN=6:CC=0
2080 FOR J=1 TO C-1
2090 R=M(J,0):CY=5:CS=CY
2100 LOCATE XN,1:COLOR7:PRINT R;"":N$(R);" (":M
(J,1);"%"
2110 GOSUB690:'レンシ"
2120 X=X+5:GOSUB720:'ワラフ
2130 X=X+20B:YN=XN+26:CC=CC+1:IF X>460 THEN HD=2
:GOSUB610:X=43:YN=6:CC=0:'リ,C
2140 NEXT J
2150 GOSUB610:'リ,C

```

```

2160 GO TO 1630
2170 ' --- ハンコウ ---
2180 WIDTH40,20:LOCATE3,3:INPUT"ハンコウ スル トワフ No
. = ",R
2190 IF R>E2 THEN BEEP:GO TO 2180
2200 LOCATE10,7:COLOR6:PRINT N$(R)
2210 LOCATES,18:COLOR6:INPUT"カクニシ...OK ? (Y/N)
",Q$:COLOR7
2220 IF Q$="Y" OR Q$="" THEN 2240
2230 IF Q$="N" THEN CLS:GO TO 2180 ELSE 2210
2240 CLS:LOCATE0,0:PRINTUSING"####:&
&":R,N
$(R)
2250 Y=3:FOR J=1 TO 10:LOCATE0,Y:PRINTUSING"####:
#####:J,Z(R,J):Y=Y+1:NEXT
2260 LOCATE0,17:INPUT"フナ"シメ / DATA ラ ハンコウ シマスカ
(0:END)= ",Q:LOCATE0,17:PRINTSPC(38)
2270 IF Q=0 THEN 60
2280 LOCATE0,17:COLOR5:PRINT"イマノ DATA =":Z(R,Q):
LOCATE20,17:COLOR7:INPUT"タタ"シイ DATA = ",Z(R,Q):L
OCATE4,Q+2:COLOR6:PRINTUSING"#####:Z(R,Q):LOCAT
E0,17:PRINTSPC(38):COLOR7:GO TO 2260
2290 ' --- ツカ ---
2300 WIDTH40,25:E2=E2+1:R=E2:DE2=E2
2310 GOSUB230:'トワフ
2320 GO TO 60
2330 ' --- ワクシ" ---
2340 WIDTH40,20
2350 LOCATES,3:INPUT"ワクシ"ヨ スル トワフ No. = ",R
2360 IF R>E2 THEN BEEP:GO TO 2350
2370 LOCATE10,7:COLOR6:PRINT N$(R)
2380 LOCATES,10:COLOR7:INPUT"ワクシ"ヨ シテ OK ? (Y/N/
N) ",Q$
2390 IF Q$="Y" OR Q$="" THEN 2410
2400 IF Q$="N" THEN CLS:GO TO 2350 ELSE 2380
2410 'N$(R)="
2420 LOCATES,15:INPUT"ツキ" カ フリマスカ ? (Y/N+リ) ",Q
$
2430 IF Q$="Y" THEN CLS:GO TO 2350
2440 IF Q$="N" OR Q$="" THEN 60 ELSE 2420
2450 ' --- 年 ノ フリマ ---
2460 WIDTH80,25:HD=0
2470 GOSUB560:'ミダ"シ
2480 X=0:Y=3:COLOR7
2490 FOR R=1 TO E2
2500 LOCATEX,Y:PRINTUSING"####: &
& ":R,N$(
R)
2510 Y=Y+1:IF Y>17 THEN Y=3:X=X+20:IF X>60 THEN
GOSUB 610:CLS:GOSUB560:X=0:'リ,C
2520 NEXT
2530 GOSUB610:'リ,C
2540 GO TO 60
2550 ' --- DATA フワ ---
2560 WIDTH40,20
2570 LOCATES,5:COLOR5:PRINT"DATA テー" "":COLOR7:
PRINT"ラ セット シテ"
2580 LOCATE10,7:COLOR6:PRINT"ロクワン "":COLOR7:PRIN
T"ニ シテワ"サイ。
2590 LOCATES,12:PRINT"( オウ"い" "":COLOR6:PRINT"RE
TURN キー "":COLOR7:PRINT"ラ オス。 )"
2600 IF INKEY$=CHR$(13) THEN 2610 ELSE 2600
2610 CLS:LOCATE7,12:COLOR5:PRINT"--- フワ"キ"ワ ---
2620 OPEN"0",#2,"CAS0:データ"
2630 FOR R=1 TO E2
2640 IF N$(R)=" " THEN 2680
2650 PRINT #2,N$(R)
2660 PRINT #2,HI(R):LO(R)
2670 PRINT #2,Z(R,1):Z(R,2):Z(R,3):Z(R,4):Z(R,5)
:Z(R,6):Z(R,7):Z(R,8):Z(R,9):Z(R,10)
2680 NEXT R:CLOSE#2
2690 LOCATE15,15:COLOR4:PRINT"カンリョウ !!":COLOR7:F
OR I=1 TO 2000:NEXT
2700 GO TO 60

```

# 拡張Kコンパイラ



COMPAC T&amp;SON

FM-8活用研究で既に『拡張Kコンパイラ』を発表しましたが、その後『FM-7用Kコンパイラはないか?』とか『コンパイラをリローケータブルにできないか?』との問い合わせが多くあったため、ここで再び発表することになりました。

上述の活用研究ではKコンパイラの特徴、展開されるオブジェクトなどについて述べましたが、ある程度マシン語をかじった人やコンパイラに慣れている人でなければ理解できなかったのではないかと思います。ここでは、実際にどうやって使うとか、効率的なデバッグなどについて記してみたいと思います。

## 概 要

多くのコンパイラがそうであるように、Kコンパイラもコンパイラ本体とランタイム・ルーチンを持っています。コンパイラ本体は実際にソース・プログラムからCPUが実行できるオブジェクトに変換するためのプログラムです。このオブジェクトはそのまま実行できるわけではなく、入出力（画面表示、キー入力など）や演算に関するものは大部分がサブルーチン・コールの形（JSR \$）に変換されます。このサブルーチン群をランタイム・ルーチンと呼びます。

コンパイル後のオブジェクトは一般にリローケータブル・バイナリと呼ばれ、これにランタイムを付けて実行できるようになったオブジェクトをエグゼキュタブル・バイナリまたはロード・モジュールと呼びます。大型計算機などのコンパイラはリローケータブル・バイナリからエグゼキュタブル・バイナリにするためにリンカやリンケージ・エディタを備えています。

Kコンパイラはそれ自身で記述されたコンパイラなので、コンパイルするときはランタイム・ルーチンが必要です。また、リローケータブル・バイナリはその名のとおりリローケータブルなので、ランタイム・ルーチンさえあればメモリ上のどこへ置いてでも実行できます。ランタイム・ルーチンはオブジェクトから一定の番地をコールされるため、リローケータはできません。これをリローケータブルにするためには前述のようなリンカやリンケージ・エディタが必要となります。

以上、コンパイラの概要について述べてみましたが、ちょっと難しかったでしょうか?ただ、コンパイラについ

て述べるのが目的ではないし、詳しく説明しようとすればかなりの量になるのでここではこれぐらいにしておきます。

## 実際の使い方

### ■プログラムの作成

KコンパイラのプログラムはBASICプログラムと同じように作成します。ただ、このときREM文の形式（行番号の後に“”を付ける）にしないとコンパイルできません。また、プログラムを打ち込むときはコンパイラは必要ありません。でき上がったプログラムはすぐにコンパイルせず、カセットやディスクにセーブしておいてください。

### ■コンパイル

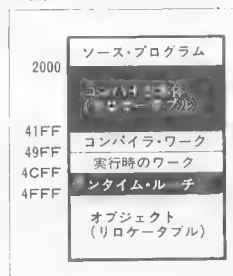
コンパイルしたいプログラムを入れた状態でCLEAR10, &H1FFFとしてください。ここで、“Out of memory”というエラーが出なければそのままコンパイルできます。リスト1, 2のコンパイラとランタイム・ルーチンをロードし、EXEC&H2000としてコンパイラを起動してください。正常にコンパイルできれば、オブジェクトの大きさを16進数で表示します。ここで一時、次のようにしてセーブしてください。

SAVEM “ファイル名”, &H4D00, &H5000+オブジェクトのサイズ, &H5000

実行アドレスはオブジェクトの先頭アドレスです。通常は5000番地ですが、オブジェクトはリローケータブルなので他の領域に移すこともできます。また、表1にもあるように207D, E番地の内容を変えれば他の番地にもオブジェクトを作れます。ただし、この番地の内容を変更するときは図1のメモリ・マップのメモリ・マップをよく見て他の領域と重ならないように注意してください。

さて、ここまではソース・プログラムが1FFF番地を越えない場合のコンパイル方法ですが、このままではあまり大きなプログラムは作れません。これ以上大きなプログラムをコンパ

図1 メモリ・マップ



イルするためには、FM 8ではファイル版への変更、FM 7では裏RAMを使う方法があります。

#### ▶ファイル版への変更 (FM-8)

ファイル版というのはソース・プログラムを外部ファイル (カセットやディスクなどの記録媒体) に入れて、そこから1行ずつ読み込んでコンパイルしようというものです。こうすると図1のメモリ・マップにあるソース・プログラムのエリアは必要なくなります。

まず、リスト3のファイル版メイン・プログラムを打ち込んでセーブしてください。次にリスト1のコンパイラをリスト4に従って変更し、次のようにしてセーブしてください。

SAVEM "KF", &H2000, &H41FF, &H2000

次にリスト2のランタイム・ルーチンをリスト5に従って変更し、次のようにしてセーブしてください。

SAVEM "RTF", &H4D00, &H4FFF, &H4D00

カセットを使っている方は、ファイル版メイン・プログラム、コンパイラ、ランタイム・ルーチンの順で1本のカセットを作れば後々便利でしょう。

さて、実際にコンパイルする手順ですが、始めにコンパイルもランタイム・ルーチンもメモリ上にロードしていない状態でソース・プログラムを打ち込みます。入力が終わったら、カセットやディスクなどにASCII形式でセーブします。実際には次のようにします。

SAVE "ファイル名", A

次に、ファイル版メイン・プログラムをRUNします。最初に "HOT OR COLD?" と聞いてくるので、コンパイラとランタイム・ルーチンが既にロードされていれば "H" を入力します (大文字で入れることを忘れずに) 。"H" 以外を入力すると外部ファイルからコンパイラとランタイム・ルーチンをロードします。

次に、"FILE NAME?" と聞いてくるので、先にASCII形式でセーブしたディスクまたはカセットを準備し、そのファイル名 (必要があればデバイス名も付けて) を入力します。この後、ソース・プログラムを読み込みながらコンパイルを始めます。

#### ■裏RAMを使う方法 (FM-7)

FM-7はソフトでBASIC ROMと裏RAMを切り換えることができるので、この裏RAMへコンパイラを移動して、大きなソース・プログラムをコンパイルすることができます (コンパイラはリロケータブルです)。

まず、リスト6、7のようなリロケータとエグゼキュータを作り、コンパイルします。リロケータはコンパイラを裏RAMへ移動するためのもので、エグゼキュータは移動したコンパイラを起動するためのものです。コンパイルは次の手順で行なってください。

- ①リロケータを使って、コンパイラを裏RAMへ移動します。アドレスの入力は必ず "\$" 記号をつけてください。

例 START ADDR = \$ 2000  
END ADDR = \$ 41FF  
DEST ADDR = \$ C000

特に、差し支えなければ上記のように入力してください。

- ②ソース・プログラムをロードします。

- ③エグゼキュータを使って、コンパイラを実行させます。

例 START ADDR = \$C000

#### ▶その他の方法

ファイル版への変更や裏RAMを使う方法については理解できたでしょうか。どちらもディスクを使っている人はあまり不便を感じないでしょうが、カセットの場合は多少、面倒です。次にその欠点を挙げてみましょう。

- ①ファイル版でコンパイル中にエラーが出ると、ソース・プログラムをロードして手直ししてセーブした後、メイン・プログラムとコンパイラを再びロードしなければならない。
- ②リロケータやエグゼキュータを使うとコンパイル中にエラーが出た場合、再度ロードし実行しなければならない。もちろん、この2つのプログラムが他の問題のない番地 (たとえば\$7C00番地や\$4000番地など) にあればいいのですが…。

そこで、FM 8でも裏RAMを使う方法とマシン語で記述したリロケータ、エグゼキュータを紹介しましょう。

リスト8はFM-7/8両用のリロケータとエグゼキュータです。プログラムはリロケータブルですが、特に問題なければリストどおりのアドレスでかまわないでしょう。

先頭のアドレスをEXECで実行すればKコンパイラを裏RAMへリロケートします。FM-8では "ディップ・スイッチ9 OFF" と "ディップ・スイッチ9 ON" とメッセージが表示されるので、指印に従ってください。次にソース・プログラムを入れて、コンパイルするときはEXEC先頭アドレス+D0 (ここではEXEC &H7CD0) としてください。FM-7ではメッセージは表示されません。これはディップ・スイッチを切り換える必要がないためです。

途中で暴走してリセットしたとしても裏RAMはクリアされないの、Kコンパイラを再びロードする必要はありません。本書のファイル版でしかコンパイルできないというプログラムは大部分、このプログラムでコンパイルできるはずですが、なお、このプログラムを使うときは一応、CLEAR10, &H41FFとしてから、ソース・プログラムを入れてください。

表1 拡張Kコンパイラ・エントリ・マップ

アドレス	内 容	機 能
2001	4C	コンパイル時作業エリア・アドレスの上位2バイト
20C0	4C	実行時変数エリア・アドレスの上位2バイト
20CE	4C	実行時スタック・エリア最終アドレスの上位2バイト
2073,4	4300	プログラム1行入カバッファ・アドレス
207D,E	5000	オブジェクト作成アドレス
2082,3	4400	ラベル・テーブル・アドレス
208E,F	4AFE	コンパイル時スタック・エリアの最終アドレス
209A,B	4B00	作業用テーブル・アドレス
4186	サブルーチン	コンパイラ1行入カルーチン
41E2	"	コンパイラOPENルーチン
41EE	"	コンパイラCLOSEルーチン
4F80	"	ランタイム1文字入カルーチン。エコーバックなし。
4F9B	"	ランタイム1文字出カルーチン。
4FB5	"	ランタイム1行入カルーチン。エコーバックあり。バック・スペースあり。

表2 エラーメッセージ

DOUBLE DEF ERROR	ラベルの二重定義がある。
ELSE WITHOUT IF	IF文がないのに、ELSEがきた。
FIWITHOUT IF	IF文がないのに、FIがきた。
ILLEGAL ARGUMENT	手続、関数のパラメータに誤りがある。
ILLEGAL EXPRESSION	許されない式である。
ILLEGAL NESTING	FOR, IF, REPEAT, WHILE句の対応がおかしい。
NEXT WITHOUT FOR	FOR文がないのに、NEXTがきた。
SYNTAX ERROR	構文に誤りがある。
TOO MANY VARIABLES	変数の数が多すぎる。
UNDEFINED LABELS	未定義のラベルである。
UNTIL WITHOUT REPEAT	REPEAT文がないのに、UNTILがきた。
WEND WITHOUT WHILE	WHILE文がないのに、WENDがきた。

## ■Kコンパイラのパログラムを入力するときの注意

Kコンパイラではコンパイル時に完全な構文チェックをしていません。ですから、変数などを使うときはプログラマーの責任で管理する必要があります。特に入力時のミスで多いのは“I”と“1”、“O”と“0”といった基本的なものです。特に変数名でこういったミスを犯すと、Kコンパイラではまったくエラーを出しません。しかも、そのまま実行してしまうこともあります。Kコンパイラのパログラムを入力するときは特に次の点に注意してください。

- ① 変数名は特に注意して入力すること。
- ② サブルーチンや関数呼び出しのパラメータは、個数が足

りなかったりしてもまったくエラーを出さないのを気を付けること。

- ③ CODE文に続く数値は、間違えると確実に暴走するので正確に入力すること。



## 最後に



説明が不十分でわかりにくかった点もあると思いますが、本書と『FM-8活用研究』を併読していただければ、Kコンパイラがかなり応用の利くコンパイラであることがわかっていただけるでしょう。

### ▶ 定数と変数

要 素	書 式	補 足
定数	1 文字定数	● 演算の中で使用する。
	文字列定数	● 関数、手続き引数中で使用する。
	10進定数	● PRINT文中で、文字列出力として使用する。
	16進定数	● -32768～32767の数値
	単純変数	● 負数は“-”をつける。
	2 バイト配列変数	● \$0000～\$FFFFの数値
	1 バイト配列変数	● 8文字まで識別する。
	%変数	● 単純変数の値を配列の先頭アドレスとして、式の値*2を加えたアドレスから始まる2バイトを指す。
		● 単純変数の値を配列の先頭アドレスとして、式の値を加えたアドレスの1バイトを指す。
		● 関数、手続きの引数の参照に用いる。
		● 数値は引数の順番を示す。
		● 局所的に領域が利用される。
ラベル	英字で始まる英数字列	● 8文字まで識別される。

### ▶ 文

処 理	書 式	補 足
一 般 処 理	変数=式 +, -, *, / >, >=, <, <=, =, <> AND, OR	代入 四則演算 関係演算 論理演算 BASICに準ずる。 演算子の優先順位は乗除、加減関係、論理
判 断	IF 条件 THEN 処理1 ELSE 処理2 FI	● 条件は式または文で記す。 ● 条件が満足されたとき、THENに対する処理を行ない、条件が満足されないとき、ELSEに対する処理を行なう。 ● ELSE部分は省略してよい。 ● IF文の終了は、FI文によって示す。
forループ	FOR 制御変数=初期値 TO 終了値 STEP 増分値 処理 NEXT	● BASICに準ずる。 ● 初期値>終了値のときは、処理を実行しない。 ● STEPを省略すると1とみなす。 ● STEPは正または負の定数。
whileループ	WHILE 条件 処理 WEND	● 条件は式または文で記す。 ● 条件が満足されている間、処理を実行する。
repeat～untilループ	REPEAT 処理 UNTIL 条件	● 条件は式または文で記す。 ● 条件が満足されるまで、処理を実行する。
サブルーチン	GOSUB ラベル名	● ラベル名のサブルーチン呼び出す。
関数、手続き	ラベル名 (引数列)	● ラベル名の関数、手続き呼び出す。 ● 引数は“, ”で区切って指定する。 ● 関数、手続き側で引数を参照するときは、%変数を用いる。 ● 関数として使用すると、関数側で最後に実行した値を持ち帰る。



## ▶文

処 理	書 式	補 足
復帰	RETURN	●サブルーチン、関数、手続きから戻る。
飛び先	GOTO ラベル名	●ラベル制御を移す。
コメント	(* コメント *)	●“(※”と“※)”でくくった部分は実行しない。
出力	PRINT 出力要素並び	●出力要素は“, ”で区切って指定する。 ●出力要素には、変数、式、文字列定数、フォーマット関数、改行子が書ける。 ●改行させるには“\n”を使用する。 ●“#”桁数、変数または式”で、桁数分右づめで10進数にして表示する。 ●フォーマット関数(\$がついている)が使える。
		●10進数、16進数(数値の前に“\$”をつける)を値とする。
入力	変数=INPUT	●押されたキーのASCIIコードを値とする。
	変数=GET	●キーが押されていないときは、0(ゼロ)を値とする。
終了	END	●実行を終了する。
宣言	CONST 名前=定数, ...	●定数に名前をつける。
	DEF ラベル名=アドレス, ...	●アドレスにラベルをつける。 ●このラベル名に対し、関数、手続き呼び出しができる。
その他	POKE アドレス, 式 CPOKE 変数名, 式の例	●アドレスに値(1バイト)を書く。 ●変数のアドレスから、式の列を書き込む。 ●変数の値は、書き込んだ分増加される。 ●式は“, ”で区切って書く。通常は1バイトとして書くが、式の後に#をつけるると2バイトとして書く。
	CODE 値の例	●値の列をプログラム中に展開する。 ●値は“, ”で区切って書く。通常は1バイトとして書くが、値の後に#をつけるると2バイトとして書く。
	CALL アドレス	●アドレスを直接サブルーチン・コールする。

## ▶関数

処 理	書 式	補 足
絶対値	ABS(式)	●式の絶対値を値とする。
アドレス値	ADR(単純変数名)	●変数のアドレスを値とする。
上位バイト	HIGH(式)	●式の上位バイトを値とする。
下位バイト	LOW(式)	●式の下位バイトを値とする。
論理否定	NOT(式)	●式の論理否定を値とする。 ●真は1, 偽は0。
アドレス内容	PEEK(アドレス)	●アドレスの内容(1バイト)を値とする。
乱数	RND(式)	●0~式-1の一樣乱数を発生する。
負数拡張	SEX(式)	●1バイトの負数を2バイトの負数とする。
符号	SGN(式)	●式の符号により、1, 0, -1の値をとる。

## ▶フォーマット関数

処 理	書 式	補 足
文字	CHR\$(式)	●式の値を文字(ASCIIコード)として出力。
16進4桁	HEX\$(式)	●式の値を16進4桁として出力。
16進2桁	HEX2\$(式)	●式の値を16進2桁として出力。
空白	SPC\$(式)	●式の値の空白を出力。

## リスト1 Kコンパイラ ダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2000	C6	4C	1F	9B	CE	44	00	BD	07	C6	00	1F	9B	7E	4F	F0	:AF
2010	BD	4D	17	BD	4D	0B	4B	20	43	4F	4D	50	49	4C	45	52	:FC
2020	20	56	45	52	53	49	4F	4E	20	31	2E	32	00	BD	4D	17	:18
2030	BD	4D	0B	43	4F	50	59	52	49	47	4B	54	20	31	39	38	:90
2040	32	20	42	59	20	43	4F	4D	50	41	43	2E	00	BD	4D	17	:0F
2050	BD	4D	17	BD	4D	17	BD	4D	0B	43	4F	4D	50	49	4C	49	:64
2060	4E	47	2E	2E	2E	00	BD	4D	17	BD	4D	17	BD	4D	17	17	:99
2070	21	70	CC	43	00	DD	00	CC	00	01	DD	02	CC	50	00	DD	:22
2080	04	CC	44	00	DD	06	CC	00	FF	9E	06	E7	84	CC	4A	FE	:E5
2090	DD	08	CC	FF	00	9E	08	ED	84	CC	4B	00	DD	0A	CC	00	:91
20A0	00	DD	0C	CC	00	DD	0E	CC	00	7F	DD	10	CC	00	00	00	:B1
20B0	DD	12	CC	00	DD	14	CC	12	12	9E	04	ED	84	CC	64	41	
20C0	4C	9E	04	ED	02	CC	1F	9B	9E	04	ED	04	CC	CE	4C	9E	:7A
20D0	04	ED	06	CC	00	9E	04	E7	08	DC	04	C3	00	09	DD	DD	
20E0	04	17	20	A2	9E	16	4F	E6	84	93	10	10	26	01	9D	17	:D8
20F0	20	FC	CC	C6	00	9E	04	ED	84	CC	1F	9B	9E	04	ED	02	:D8
Sum:	F0	C1	B7	60	D5	2D	91	39	13	B6	E5	04	8E	54	BB	3D	:F0

21E0	BD	4D	17	BD	4D	0B	55	4E	44	45	46	49	4E	45	44	20	:E8
21F0	4C	41	42	45	4C	53	20	3A	00	BD	4D	17	BD	4D	0B	20	:63
Sum:	BB	DE	C7	1A	B3	2A	AE	8A	BB	44	12	DA	D5	9D	A2	05	:33
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2200	00	CC	00	04	DD	1C	CC	00	0B	36	06	DC	1C	16	00	11	:FB
2210	DC	1C	9E	1B	E6	B8	4F	BD	4E	DE	DC	1C	C3	00	01	DD	:F0
2220	1C	A3	C4	2F	EB	33	42	CC	00	01	DD	14	DC	1B	C3	00	:BD
2230	0C	DD	1B	16	FF	87	BD	4D	17	BD	4D	17	BD	4D	0B	4F	:43
2240	42	4A	45	43	54	20	50	52	4F	47	52	41	4D	20	53	49	:5C
2250	5A	45	20	3A	00	DC	04	B3	4F	FF	DD	1E	DC	1E	BD	4E	:AA
2260	B0	BD	4D	0B	20	2B	24	00	DC	1E	BD	4E	BF	BD	4D	0B	:A4
2270	29	00	BD	4D	17	BD	4D	17	BD	4D	0B	43	4F	4D	50	49	:FB
2280	4C	45	20	45	4E	44	2E	00	BD	4D	17	39	17	12	33	9E	:0A
2290	16	4F	E6	84	36	06	CC	00	3B	BD	4D	B9	36	06	9E	16	:C5
22A0	4F	E6	84	36	06	CC	00	3B	BD	4D	B9	36	06	9E	16	16	:C1
22B0	27	00	0A	DC	16	C3	00	01	DD	16	16	FF	CF	9E	16	4F	:C1
22C0	E6	84	93	0E	10	26	00	03	1E	16	DC	14	36	06	CC	:66	
22D0	00	01	BD	4D	B9	36	06	30	BC	02	20	03	2B	2A	00	3A	:67
22E0	10	17	10	EE	32	62	AA	00	EA	00	10	27	00	5D	CC	00	:2D
22F0	01	DD	14	DC	16	C3	00	01	DD	16	16	9E	16	4F	E6	1F	:36
Sum:	4B	A7	F1	36	E9	9C	89	F2	A2	C6	1A	DD	16	06	14	71	:09
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2300	06	CC	00	2A	BD	4D	B9	36	06	9E	16	4F	E6	84	36	06	:A4
2310	CC	00	29	BD	4D	B9	4A	C0	E4	C0	36	06	9E	16	4F	E6	:E5
2320	1F	36	06	DC	0E	BD	4D	B9	4A	C0	EA	C0	27	C5	9E	16	:BC
2330	4F	E6	1F	93	0E	10	26	00	03	16	FF	AD	56	DC	16	C3	:00
2340	01	DD	16	CC	00	00	DD	14	16	FF	41	30	BC	02	20	06	:EB
2350	50	52	49	4E	54	00	34	10	17	10	77	32	62	36	06	30	:6F
2360	8C	02	20	02	3F	00	34	10	17	10	67	32	62	AA	C0	EA	:A9
2370	C0	10	27	01	BF	17	11	4A	9E	16	4F	E6	84	36	06	CC	:45
2380	10	26	00	0A	DC	16	C3	00	01	DD	16	16	FF	E7	9E	16	:99

## リスト1 Kコンパイラ ダンプ・リスト

2390	4F	E6	84	83	00	2F	10	26	00	14	CC	4D	17	34	06	17	:36
23A0	10	84	32	62	DC	16	C3	00	01	DD	16	16	FF	C7	9E	16	:61
23B0	4F	E6	84	83	00	23	10	26	00	3E	DC	16	C3	00	01	DD	:66
23C0	16	17	11	75	9E	16	4F	E6	84	83	00	2C	10	27	00	03	:09
23D0	16	1C	5B	CC	36	06	9E	04	ED	84	DC	04	C3	00	02	DD	:2A
23E0	04	DC	16	C3	00	01	DD	16	17	11	4E	CC	4E	84	34	06	:01
23F0	17	10	33	32	62	16	FF	7D	9E	16	4F	E6	84	83	00	22	:92
Sum:	E2	BE	E3	1B	66	9B	95	F6	A1	A3	EE	A5	DB	F0	45	76	:84
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2400	10	26	00	61	CC	4D	0B	34	06	17	10	1A	32	62	DC	16	:8C
2410	C3	00	01	DD	16	9E	16	4F	E6	84	9E	04	E7	84	DC	04	:11
2420	C3	00	01	DD	04	9E	16	4F	E6	84	36	06	CC	00	22	BD	:F9
2430	4D	B9	36	06	9E	16	4F	E6	84	36	06	DC	0E	BD	4D	B9	:F8
2440	AA	04	EA	0C	27	C8	CC	00	00	9E	04	E7	1F	9E	16	4F	:7A
2450	EA	84	93	0E	10	26	00	03	16	FC	86	DC	16	C3	00	01	:92
2460	DD	16	16	FF	10	30	CC	02	20	06	43	48	52	24	28	00	:25
2470	34	10	17	0F	5D	32	62	10	27	00	10	17	10	9F	CC	4E	:82
2480	DE	34	06	17	0F	0A	32	62	16	FE	EA	30	BC	02	20	06	:54
2490	48	45	58	24	28	00	34	10	17	0F	37	32	62	10	27	00	:9D
24A0	10	17	10	79	CC	4E	BF	34	06	17	0F	7A	32	62	16	FE	:07
24B0	C4	30	BC	02	20	07	48	45	58	32	24	28	00	34	10	17	:6B
24C0	0F	10	32	62	10	27	00	10	17	52	CC	4E	C5	34	06	:8C	
24D0	17	0F	5D	32	62	16	FE	9D	30	BC	02	20	06	53	50	43	:88
24E0	24	28	00	34	10	17	0E	EA	32	62	10	27	00	10	17	10	:A1
24F0	2C	CC	4E	B6	34	06	17	0F	2D	32	62	16	FE	77	9E	16	:5C
Sum:	F4	1C	AF	31	01	3E	DD	5E	E4	7B	E1	4F	FC	0E	D7	BB	:85
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2500	4F	E6	84	36	06	CC	00	38	BD	4D	B9	36	06	9E	16	4F	:FE
2510	E6	84	36	06	DC	0E	BD	4D	B9	AA	0C	EA	0C	10	27	00	:9E
2520	03	16	F0	68	17	10	12	CC	4E	B0	34	06	17	0E	F7	32	:06
2530	62	16	FE	41	30	BC	02	20	06	43	4F	4E	53	54	00	34	:56
2540	10	17	0E	BE	32	62	10	27	00	08	CC	03	00	DD	20	16	:78
2550	00	19	30	CC	02	20	04	44	45	46	00	34	10	17	0E	72	:A5
2560	32	62	10	27	00	9B	CC	02	00	DD	20	DC	16	B3	00	01	:A7
2570	DD	16	DC	16	C3	00	01	DD	16	17	0D	17	DC	22	B3	00	:58
2580	03	10	27	00	1B	BD	4D	17	BD	4D	0E	44	4F	55	42	4C	:01
2590	45	20	44	45	46	20	45	52	52	4F	52	2E	00	16	1A	AB	:E4
25A0	9E	16	4F	E6	84	83	00	3D	10	27	00	03	16	CC	DC	DC	:32
25B0	08	B3	00	02	DD	0B	DC	18	9E	08	ED	84	DC	16	C3	00	:32
25C0	01	DD	16	17	0F	73	9E	08	ED	84	DC	18	DC	08	C3	00	:3F
25D0	02	DD	0B	DC	1A	83	00	01	10	27	00	03	16	CC	9D	DC	:36
25E0	20	9E	16	ED	84	DC	24	9E	18	ED	02	DC	04	B3	00	03	:52
25F0	DD	04	9E	16	4F	E6	84	83	00	2C	10	27	FF	74	16	FC	:B9
Sum:	A7	63	6D	5F	DE	B3	66	A6	F6	BB	2E	B5	6B	41	47	E9	:E0
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2600	B8	30	BC	02	20	06	47	4F	5A	4F	20	00	34	10	17	0D	:30
2610	C1	32	62	10	27	00	17	17	0C	4F	79	CC	00	16	9E	04	:A0
2620	84	DC	04	C3	00	03	DD	04	17	0E	AE	16	FC	5E	30	BC	:0A
2630	02	20	07	47	4F	53	55	42	20	00	34	10	17	0D	93	32	:F6
2640	62	10	27	02	07	17	0C	4B	17	00	03	16	FC	3E	DC	08	:57
2650	B3	00	04	DD	0B	DC	18	9E	08	ED	84	DC	22	9E	08	ED	:08
2660	02	CC	00	06	DD	26	9E	16	4F	E6	84	B3	00	5B	10	26	:52
2670	01	48	9E	16	4F	E6	01	B3	00	5D	10	26	00	0A	DC	16	:45
2680	C3	00	02	DD	16	16	01	32	DC	16	C3	00	01	DD	16	DC	:86
2690	26	C3	00	01	DD	26	9E	16	4F	E6	84	B3	00	22	10	26	:35
26A0	00	A5	CC	30	BC	9E	04	ED	84	CC	02	20	9E	04	ED	02	:BF
26B0	DC	04	C3	00	05	DD	04	DC	04	DD	2B	CC	00	00	DD	2A	:41
26C0	DC	16	C3	00	01	DD	16	DC	2A	C3	00	01	DD	2A	9E	16	:2E
26D0	4F	E6	84	9E	04	E7	84	DC	04	C3	00	01	DD	04	9E	16	:FF
26E0	4F	E6	84	36	06	CC	00	22	BD	4D	B9	36	06	9E	16	4F	:E5
26F0	E6	84	36	06	DC	0E	BD	4D	B9	AA	0C	EA	0C	27	C1	DC	:2B
Sum:	DF	54	54	F9	35	B0	51	66	5C	2B	D3	52	9A	50	B1	68	:CB
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2700	2A	36	06	CC	00	7F	BD	4D	A9	36	06	9E	16	4F	E6	84	:0D
2710	36	06	DC	0E	BD	4D	B9	AA	0C	EA	0C	10	27	00	03	16	:4D
2720	19	7E	CC	00	00	9E	04	E7	1F	DC	0A	9E	2B	E7	1F	CC	:A9
2730	36	10	9E	04	ED	84	DC	16	C3	00	01	DD	16	DC	04	C3	:A5
2740	00	02	DD	04	16	00	11	17	0D	EF	CC	36	06	9E	04	ED	:B4
2750	84	DC	04	C3	00	02	DD	04	9E	16	4F	E6	84	B3	00	2C	:26
2760	10	27	FF	24	9E	16	4F	E6	84	B3	00	5D	10	27	00	03	:E1
2770	16	17	2D	DC	16	C3	00	01	DD	16	DC	26	B3	00	00	10	:9A
2780	2F	00	37	CC	00	34	9E	04	E7	1E	CC	00	02	DD	1C	DC	:B0
2790	26	36	06	DC	1C	16	00	1C	CC	37	10	9E	04	ED	84	CC	:7E
27A0	34	10	9E	04	ED	02	DD	04	C3	00	04	DD	04	DC	1C	C3	:18
27B0	00	01	DD	1C	A3	C4	2F	0E	C3	42	CC	00	17	9E	04	E7	:18
27C0	84	DC	04	C3	00	03	DD	04	9E	08	ED	84	DC	1B	9E	08	:BC
27D0	EC	02	DD	22	CC	0B	DC	03	00	4D	08	9E	1B	4F	E6	84	:EC
27E0	B3	00	02	10	27	00	06	17	0C	EF	16	0F	CC	00	BD	:82	
27F0	9E	04	E7	1D	9E	18	EC	02	9E	04	ED	1E	DC	26	B3	00	:7C
Sum:	73	11	DB	7F	C1	FC	CE	17	4C	09	BB	83	99	F7	D7	F0	:3A
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2800	00	10	2F	00	3F	CC	00	32	9E	04	E7	84	DC	04	C3	00	:2C
2810	02	DD	04	DC	26	B3	00	08	10	2C	00	12	CC	00	06	36	:20
2820	06	DC	26	58	49	E3	C1	9E	04	E7	1F	16	00	16	CC	00	:ED
2830	E9	9E	04	E7	1F	DC	26	58	49	9E	04	ED	84	DC	04	C3	:EA
2840	00	02	DD	04	39	30	BC	02	07	52	45	54	55	52	4E	:E1	
2850	00	34	10	17	0B	37	32	62	10	27	00	11	CC	00	39	9E	:61
2860	04	E7	84	DC	04	C3	00	01	DD	04	16	FA	1F	30	BC	02	:E1
2870	20	07	52	45	50	45	41	54	00	34	10	17	0B	54	32	62	:36
2880	10	27	00	1D	DC	08	B3	00	0A	DD	0B	DC	12	9E	08	ED	:2B
2890	08	CC	FF	FD	9E	08	ED	84	DC	04	9E	08	ED	02	16	F9	:68
28A0	EB	30	BC	02	20	07	55	4E	54	49	4C	20	00	34	10	17	:D7
28B0	0B	20	32	62	10	27											

28C0	27	00	1F	BD	4D	17	BD	4D	0B	55	4E	54	49	4C	20	57	:7F
28D0	49	54	48	4F	55	54	20	52	45	50	45	41	54	2E	00	16	:02
28E0	FB	33	17	0C	54	17	17	0F	9E	08	ED	02	93	04	B3	FF	:6D

## リスト 1 Kコンパイル ダンプ・リスト

2DFO	16	04	B9	DC	24	DD	34	CC	00	9E	9E	04	E7	1E	17	06	:E2	
Sum:	75	AB	E5	FF	F3	94	20	29	42	FA	BA	9D	7F	B5	22	23	:B0	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
2E00	C1	DC	16	C3	00	01	DD	16	17	07	2E	9E	16	4F	E6	B4	:23	
2E10	B3	00	23	10	26	00	14	CC	ED	B1	9E	04	ED	B4	DC	16	:2F	
2E20	C3	00	01	DD	16	17	06	14	CC	ED	B1	9E	04	ED	B4	DC	:16	
2E30	ED	B4	DC	04	C3	00	02	DD	04	9E	16	4F	EF	E6	B4	83	00	:E7
2E40	2C	27	BE	CC	00	9F	9E	04	E7	BA	DC	34	9E	04	E7	01	:23	
2E50	DC	04	C3	00	02	DD	04	16	F4	32	30	BC	02	20	05	43	:E8	
2E60	41	4C	4C	00	34	10	17	05	69	32	62	10	27	00	2E	17	:B2	
2E70	06	C7	DC	1A	B3	00	01	10	26	00	0A	CC	00	BD	9E	04	:B2	
2E80	E7	1D	16	F4	07	CC	1F	01	9E	04	ED	BA	CC	AD	84	9E	:AF	
2E90	04	ED	02	DC	04	C3	00	04	DD	04	16	F3	EF	30	BC	02	:31	
2EA0	20	05	43	4F	44	45	00	34	10	17	05	26	32	62	10	27	:91	
2EB0	00	60	DC	16	B3	00	01	DD	16	DC	16	C3	00	01	DD	16	:72	
2EC0	17	05	FF	17	0A	AO	DC	1A	B3	00	01	10	27	00	03	16	:A6	
2ED0	03	AA	9E	16	4F	E6	B4	83	00	23	10	26	00	17	DC	04	:E8	
2EE0	B3	00	01	DD	04	DC	24	9E	04	ED	1E	DC	16	C3	00	01	:CB	
2EF0	DD	16	16	00	0D	DC	04	B3	00	02	DD	04	DC	24	9E	04	:FE	
Sum:	C8	D2	AA	D9	F4	B6	5B	5C	B0	1B	BE	CF	9D	F6	15	F9	:44	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
2F00	E7	1F	17	05	BD	9E	16	4F	E6	B4	83	00	2C	27	AA	16	:E2	
2F10	F3	7A	30	BC	02	20	04	45	4E	4A	00	34	10	17	04	B2	:37	
2F20	32	62	10	27	00	26	CC	C6	00	9E	04	ED	BA	CC	1F	98	:1C	
2F30	9E	04	ED	02	CC	7E	4F	9E	04	ED	04	CC	00	F0	9E	04	:1B	
2F40	E7	06	DC	04	C3	00	07	DD	04	16	F3	40	9E	16	4F	E6	:A4	
2F50	B4	34	06	17	05	02	32	62	10	27	02	9C	17	03	34	9E	:31	
2F60	16	4F	E6	B4	83	00	5B	10	26	00	06	17	F6	E0	16	F3	:DF	
2F70	1B	9E	16	4F	E6	B4	83	00	3B	10	26	00	6F	DC	22	83	:6A	
2F80	00	03	10	26	00	10	CC	00	00	9E	1B	ED	BA	CC	04	9E	:BA	
2F90	1B	ED	02	16	F2	F6	9E	1B	ED	4F	E6	01	B3	00	10	26	:AA	
2FA0	00	1B	BD	4D	17	BD	4D	0B	44	4F	55	42	4C	45	20	44	:70	
2FB0	45	46	20	45	52	52	4F	52	2E	00	16	10	BB	9E	1B	ED	:B6	
2FC0	02	DD	36	CC	00	00	9E	1B	ED	01	9E	36	EC	1E	DD	2E	:68	
2FD0	DC	04	93	36	9E	36	ED	1E	DC	2E	DD	36	DC	36	83	00	:3A	
2FE0	00	26	E7	DC	04	9E	1B	ED	02	16	F2	AO	DC	22	83	00	:BB	
2FF0	03	10	26	00	20	CC	01	00	9E	1B	ED	BA	CC	0C	9E	1B	:EB	
Sum:	B4	BE	E7	54	D9	D7	F6	DF	D1	D0	BA	32	B5	10	F3	9B	:4B	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
3000	ED	02	DC	0C	C3	00	02	DD	0C	DC	83	01	02	10	26	:29		
3010	00	03	16	10	A6	9E	1B	ED	02	DD	3B	9E	16	4F	E6	B4	:F5	
3020	B3	00	3A	10	26	00	5B	DC	16	C3	00	01	DD	16	17	05	:10	
3030	0B	DC	1A	B3	00	01	10	26	00	0B	CC	00	E7	DD	3A	16	:A0	
3040	00	AE	CC	00	9E	04	E7	BA	DC	3B	9E	04	E7	01	CC	:BF		
3050	30	BB	9E	04	ED	02	CC	36	10	9E	04	ED	04	DC	04	C3	:94	
3060	00	06	DD	04	DC	16	C3	00	02	DD	16	17	04	CB	CC	E7	:2A	
3070	D1	9E	04	ED	BA	DC	04	C3	00	02	DD	04	16	F2	0D	9E	:1D	
3080	16	4F	E6	B4	83	00	2B	10	26	01	40	DC	16	C3	00	01	:A7	
3090	DD	16	17	04	AA	DC	1A	B3	00	01	10	26	00	0E	CC	00	:3C	
30A0	ED	DD	3A	DC	24	5B	49	DD	24	16	00	44	CC	5B	49	9E	:0B	
30B0	04	ED	BA	CC	00	9E	04	E7	02	DC	3B	9E	04	E7	03	:0A		
30C0	CC	30	BB	9E	04	ED	04	CC	36	10	9E	04	ED	06	DC	04	:A1	
30D0	C3	00	0B	DD	04	DC	16	C3	00	02	DD	16	17	04	5A	CC	:97	
30E0	ED	D1	9E	04	ED	BA	DC	04	C3	00	02	DD	04	16	F1	9C	:FA	
30F0	DC	04	B3	00	03	DD	04	DC	0B	B3	00	02	DD	0B	DC	24	:95	
Sum:	B5	F2	00	53	BD	2D	3C	BE	EC	BC	EB	3F	62	19	24	0B	:F7	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
3100	9E	08	ED	B4	DC	16	C3	00	02	DD	16	17	07	2B	CC	00	:D3	
3110	9E	0E	04	E7	BA	DC	3B	9E	04	E7	01	DC	3A	9E	04	E7	:E8	
3120	02	9E	04	ED	BA	DC	24	0B	C3	00	02	DD	0B	DC	04	E7	:A7	
3130	B3	00	00	10	26	00	11	CC	00	BA	9E	04	E7	03	DC	04	:B6	
3140	C3	00	04	DD	04	16	F1	4A	DC	24	36	06	CC	00	10	BD	:CB	
3150	4D	A1	36	06	DC	24	36	06	CC	FF	ED	BD	4D	A9	A4	C0	:37	
3160	E4	C0	10	27	00	19	DC	24	36	06	CC	00	1F	A4	C0	E4	:63	
3170	C0	9E	04	E7	03	DC	04	C3	00	04	DD	04	16	F1	0D	DC	:C4	
3180	24	36	06	CC	00	BD	4D	A1	36	06	DC	24	36	06	CC	:9B		
3190	FF	7F	BD	4D	A9	A4	C0	E4	C0	10	27	00	17	CC	00	BB	:D8	
31A0	9E	04	E7	03	DC	24	9E	04	E7	04	DC	04	C3	00	05	DD	:9E	
31B0	04	16	F0	DD	CC	00	89	9E	04	E7	03	DC	24	9E	04	ED	:52	
31C0	04	DC	04	C3	00	06	DD	04	16	F0	C1	9E	16	4F	E6	B4	:C2	
31D0	B3	00	3D	10	26	00	21	DC	16	C3	00	01	DD	16	17	03	:DA	
31E0	58	CC	00	DD	9E	04	E7	BA	DC	3B	9E	04	E7	01	DC	04	:BC	
31F0	C3	00	02	DD	04	16	F0	94	9E	16	4F	E6	B4	83	00	25	:55	
Sum:	DC	BA	24	D9	06	66	B0	42	DE	6A	3D	05	D0	9B	F1	1A	:F1	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
3200	10	26	00	7B	DC	16	C3	00	01	DD	16	17	07	5B	9E	16	:B1	
3210	4F	E6	B4	36	06	CC	00	3D	BD	4D	C1	36	06	DC	1A	36	:31	
3220	06	CC	00	01	BD	4D	C1	AA	C0	EA	C0	10	27	00	03	16	:02	
3230	00	4A	DC	04	B3	00	03	DD	04	DC	24	5B	49	DD	3C	DC	:27	
3240	16	C3	00	01	DD	16	17	02	F0	DC	3C	B8	00	10	10	2C	:BD	
3250	00	13	CC	ED	60	D3	3C	9E	04	ED	BA	DC	04	C3	00	02	:F3	
3260	DD	04	16	00	14	CC	ED	E8	9E	04	ED	BA	DC	3C	9E	04	:79	
3270	E7	02	DC	04	C3	00	03	DD	04	16	F0	10	BD	4D	17	BD	:64	
3280	4D	0B	53	49	4E	54	A1	5B	20	45	52	4F	52	2E	00	:17		
3290	16	0D	B5	17	02	2C	9E	16	4F	E6	B4	34	06	17	01	BB	:94	
32A0	32	62	B3	00	00	12	12	10	26	00	03	16	FF	CC	CC	:34		
32B0	00	00	DD	1C	CC	00	07	36	06	DC	1C	16	00	13	DC	1C	:21	
32C0	10	9E	04	A1	AB	CC	00	20	E7	A4	DC	1C	C3	00	01	DD	:1A	
32D0	1C	A3	C4	2F	E9	33	42	CC	00	DD	22	9E	16	4F	E6	:C4		
32E0	B4	34	06	17	01	72	32	62	36	06	9E	16	4F	E6	B4	:39		
32F0	06	17	01	47	32	62	AA	C0	EA	C0	10	27	00	2F	DC	22	:71	
Sum:	BA	04	5B	EF	19	49	E0	ED	A4	6A	B1	C2	35	13	44	E6	:FA	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3300	B3	00	0B	10	2C	00	1C	DC	22	10	9E	0A	31	AB	9E	16	:29
3310	4F	E6	B4	34	06	17	01	7D	32	62	E7	A4	DC	22	C3	00	:6B
3320	01	DD	22	DC	16	C3	00	01	DD	16	16	FF	AF	CC	00	00	:39
3330	DD	22	DC	06	DD	1B	1E	4F	E6	B4	83	00	FF	10	26	:FD	
3340	00	4B	CC	00	00	DD	1C	CC	00	03	36	06	DC	1C	16	00	:26
3350	1D	CC	1C	C3	00	02	5B	49	10	9E	1B	31	AB	DC	1C	9E	:18
3360	0A	5B	49	EC	8B	ED	A4	DC	1C	C3	00	01	DD	1C	A3	C4	:F1
3370	2F	DD	32	42	CC	00	00	9E	1B	E7	84	CC	00	FF	9E	18	:C1
3380	E7	CC	00	00	03	DD	22	16	10	3C	CC	00	00	DD	1C	DE	:1B
3390	1C	C3	00	04	9E	1B	E6	8B	4F	CC	36	06	DC	1C	9E	0A	:84
33A0	BE	4F	DD	1B	10	27	00	00	0A	DC	1B	C3	00	00	DD	1B	:9E
33B0	16	00	13	DC	1C	C3	00	01	DD	1C	DC	1C	C3	00	08	26	:87
33C0	CE	CC	00	1D	DD	22	DC	22	B3	00	00	10	2F	FF	67	16	:D6
33D0	00	F0	EC	62	DD	2E	CC	00	00	DD	1C	DC	1C	9E	2E	E6	:88
33E0	BE	4F	B3	00	10	27	00	2E	DC	1C	9E	2E	E6	B8	4F	:46	
33F0	36	06	DC	1C	9E	1E	16	EB	8B	4F	CC	36	06	DC	1C	9E	:22
Sum:	39	6F	D5	C3	52	FC	B7	50	FA	10	F5	90	38	4C	41	63	:4C
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3400	BD	4D	C1	10	27	00	06	CC	00	00	DD	2E	39	DC	1C	C3	:D3
3410	00	01	DD	1C	16	FF	0A	DC	16	D3	1C	DC	16	17	00	A2	:60
3420	CC	00	01	DD	2E	39	CC	00	DD	9E	04	E7	84	DC	04	C3	:4A
3430	00	03	DD	04	EC	62	9E	04	ED	1E	39	EC	62	36	06	CC	:6E
3440	00	30	BD	4D	99	36	06	EC	62	36	06	CC	00	39	BD	4D	:AB
3450	B1	A4	C0	E4	C0	DD	2E	39	EC	62	36	06	CC	00	41	BD	:51
3460	4D	99	36	06	EC	62	36	06	CC	00	5A	BD	4D	B1	A4	C0	:F6
3470	E4	C0	36	06	EC	62	36	06	CC	00	B1	BD	4D	99	36	06	:71
3480	EC	62	36	06	CC	00	7A	BD	4D	B1	A4	C0	E4	C0	A4	C0	:FD
3490	EA	C0	DD	2E	39	EC	62	36	06	CC	00	60	BD	4D	A9	36	:8D
34A0	06	EC	62	36	06	CC	00	7B	BD	4D	B1	A4	C0	E4	C0	10	:9A
34B0	27	00	0A	EC	62	83	00	20	DD	40	16	00	04	EC	62	DD	:84
34C0	40	39	9E	16	4F	E6	B4	83	00	20	10	26	00	0A	DC	16	:8B
34D0	C3	00	01	DD	16	16	FF	EA	39	DC	22	B3	00	03	10	26	:A9
34E0	00	15	CC	00	00	9E	04	ED	1E	CC	00	01	9E	1B	ED	B4	:82
34F0	DC	04	9E	1B	ED	02	39	9E	1B	4F	E6	01	B3	00	10	:3D	
Sum:	4D	DE	ED	AB	47	4B	70	63	02	4B	0A	99	21	BA	4C	77	:16
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3500	26	00	0B	9E	1B	EC	02	93	04	9E	04	ED	1E	39	9E	1B	:08
3510	EC	02	9E	04	ED	1E	DC	0A	9E	1B	ED	02	3E	17	00	19	:89
3520	9E	16	4F	E6	B4	83	00	29	10	27	00	03	16	0A	FF	DC	:4E
3530	16	C3	00	01	DD	16	16	FF	B9	17	FF	B6	17	00	A3	9E	:5F
3540	16	4F	E6	B4	DD	36	30	8C	02	20	04	41	4E	44	00	34	:C8
3550	10	17	FE	7E	32	62	36	06	30	8C	02	20	03	4F	52	00	:F5
3560	34	10	17	FE	DD	32	62	AA	C0	EA	C0	10	27	00	70	DC	:F1
3570	0B	83	00	02	DD	0B	DC	36	9E	0B	ED	84	CC	36	06	9E	:4A
3580	04	ED	B4	DC	04	C3	00	02	DD	04	17	00	55	9E	0B	EC	:FB
3590	B4	DD	2E	DC	0B	C3	00	02	DD	0B	DC	2E	B3	00	41	10	:F8
35A0	26	00	0B	CC	00	84	DD	2E	16	00	05	CC	00	8A	DD	2E	:05
35B0	DC	2E	C3	00	20	9E	04	E7	B4	CC	00	C0	9E	04	E7	01	:10
35C0	DC	2E	C3	00	60	9E	04	E7	02	CC	00	C0	9E	04	E7	03	:D0
35D0	DC	04	C3	00	04	DD	04	CC	00	05	DD	1A	16	FF	67	16	:E2
35E0	FE	E0	17	FE	DD	17	00	E7	30	8C	02	20	03	C3	3E	00	:29
35F0	34	10	17	FD	DD	32	62	10	27	00	08	CC	00	4C	C1	DD	:EB
Sum:	9C	EE	24	0A	09	E1	E3	F4	7B	C7	B2	ED	42	4F	7E	C9	:FF
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3600	16	00	81	30	BC	02	00	03	C3	3D	00	34	10	17	FD	DC	:0B
3610	32	62	10	27	00	08	CC	4D	B1	DD	2C	16	00	66	30	BC	:DE
3620	02	20	03	C3	3D	00	04	10	17	FD	AB	32	62	10	27	00	:6A
3630	0B	CC	4D	99	DD	2C	16	00	4B	30	8C	02	20	02	30	00	:41
3640	34	10	17	FD	DD	32	62	10	27	00	0B	CC	4D	B9	DD	2C	:93
3650	16	00	31	30	BC	02	00	20	02	3E	00	34	10	17	FD	32	:62
3660	62	10	27	00	0B	CC	4D	A9	DD	2C	16	00	17	30	8C	02	:57
3670	20	02	3C	00	CC	10	17	FD	59	32	62	10	27	00	4D	CC	:F3
3680	4D	A1	DD	2C	DC	0B	83	00	02	DD	0B	DC	2C	9E	0B	ED	:E0
3690	B4	CC	36	06	9E	04	ED	B4	DC	04	C3	00	02	DD	04	17	:3C
36A0	00	2D	9E	0B	EC	84	DC	2C	DC	0B	C3	00	02	DD	0B	CC	:A6
36B0	00	BD	9E	04	E7	B4	DC	04	C3	00	03	DD	04	DC	2C	9E	:F7
36C0	04	ED	1E	DC	1A	DD	42	CC	00	04	DD	1A	16	FD	F3	17	:08
36D0	FD	F0	17	01	21	9E	16	4F	E6	B4	36	06	CC	00	2B	BD	:83
36E0	4D	B9	36	06	9E	16	4F	E6	B4	36	06	CC	00	02	DD	0B	:4E
36F0	B9	AA	C0	EA	C0	10	27	00	FA	DC	0B	83	00	02	DD	0B	:4C
Sum:	F6	07	06	66	E1	F8	13	CD	CB	2B	C5	92	4A	D5	B2	11	:51
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3700	9E	16	4F	E6	B4	9E	0B	ED	B4	DC	04	C3	00	02	DD	0B	:8B
3710	DC	C5	9E	0B	EC	B4	DD	2E	DC	0B	C3	00	02	DD	0B	DC	:8E
3720	B3	00	2B	10	26	00	51	DC	1A	83	00	01	10	26	00	17	:FC
3730	DC	04	B3	00	02	DD	04	CC	00	C3	9E	04	E7	1D	DC	24	:7B
3740	DC	04	ED	1E	16	10	2E	DC	1A	83	00	02	10	26	00	17	:89
3750	9E	04	B3	00	02	DD	04	CC	00	D3	9E	04	E7	1E	DC	24	:8C
3760	9E	04	E7	1F	16	10	0E	CC	E3	C1	9E	04	ED	B4	DC	04	:CF
37B0	C3	00	02	DD	04	16	00	63	DC	1A	83	00	01	10	26	00	:0A
3790	17	DC	04	B3	00	02	DD	04	CC	00	B3	9E	04	E7	1D	DC	:2E
37A0	24	9E	04	ED	1E	16	10	00	43	DC	1A	83	00	02	10	26	:0B
37B0	17	DC	04	B3	00	02	DD	04	CC	00	93	9E	04	E7	1E	DC	:13
37C0	24	9E	04	E7	1F	16	10	00	23	CC	A3	C1	9E	04	ED	B4	:CC
37D0	53	43	9E	04	ED	02	CC	C3	00	9E	04	ED	04	CC	00	01	:16
37E0	9E	04	E7	06	DC	04	C3	00	07	DD	04	CC	00	05	DD	1A	:E2
37F0	16	FE	E2	16	FC	CC	17	FC	C9	17	01	6A	9E	16	4F	E6	:1B
Sum:	F6	33	DB	94	68	3A	AF	51	93	2B	D6	D3	46	DA	9B	43	:9C
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3800	B4	36	06	CC	00	2A	BD	4B	B9	AA	C0	EA	C0	10	27	01	:83
3810	36	06	CC	00	2F	BD	4D	B9	AA	C0	EA	C0	10	27	01	03	:89
3820	DC	0B	83	00	02	DD	0B	9E	16	4F	E6	B4	9E	0B	ED	B4	:D0

## リスト1 Kコンパイル ダンプ・リスト

3830	DC	16	C3	00	01	DD	16	CC	36	06	9E	04	ED	B4	DC	04	:A4
3840	C3	00	02	DD	04	17	01	1E	9E	08	EC	B4	DD	2E	DC	08	:E1
3850	C3	00	02	DD	08	DC	2E	B3	00	2A	10	26	00	80	DC	1A	:00
3860	36	06	CC	00	01	BD	4D	B9	36	06	DC	24	36	06	CC	00	:10
3870	02	BD	4D	B9	36	06	DC	24	36	06	CC	00	04	BD	4D	B9	:D0
3880	AA	C0	EA	C0	36	06	DC	24	36	06	CC	00	08	BD	4D	B9	:23
3890	AA	C0	EA	C0	36	06	DC	24	36	06	CC	00	10	BD	4D	B9	:18
38A0	AA	C0	EA	C0	EA	C0	E4	04	10	27	00	25	DC	04	B3	00	:DB
38B0	05	DD	04	CC	58	49	9E	04	ED	B4	DC	04	C3	00	02	DD	:E8
38C0	04	DC	24	47	56	DD	24	DC	24	B3	00	01	26	E5	16	00	:47
38D0	0A	CC	4D	1F	34	06	17	FB	4D	32	62	16	00	7D	DC	1A	:F8
38E0	36	06	CC	00	01	BD	4D	B9	36	06	DC	24	36	06	CC	00	:10
38F0	02	BD	4D	B9	36	06	DC	24	36	06	CC	00	04	BD	4D	B9	:D0
Sum: 79 A5 B1 6A 9E 12 1E AE FF 01 96 1B DF 16 AB 4C :1F																	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3900	AA	C0	EA	C0	36	06	DC	24	36	06	CC	00	08	BD	4D	B9	:23
3910	AA	C0	EA	C0	36	06	DC	24	36	06	CC	00	10	BD	4D	B9	:28
3920	AA	C0	EA	C0	EA	C0	E4	04	10	27	00	25	DC	04	B3	00	:DB
3930	05	DD	04	CC	47	56	9E	04	ED	B4	DC	04	C3	00	02	DD	:E4
3940	04	DC	24	47	56	DD	24	DC	24	B3	00	01	26	E5	16	00	:47
3950	0A	CC	4D	3B	34	06	17	FB	4D	32	62	CC	00	05	DD	1A	:D2
3960	16	FE	99	16	FB	5C	30	8C	02	20	06	49	4E	50	55	54	:8E
3970	00	34	10	17	FA	5C	32	62	10	27	00	12	CC	4D	E2	34	:1D
3980	06	17	FA	A2	32	62	CC	00	03	DD	1A	16	FA	34	30	8C	:14
3990	02	0A	4D	47	45	54	00	34	10	17	FA	36	32	62	10	27	:5C
39A0	00	12	CC	4D	1F	34	06	17	FB	4D	32	62	CC	00	03	DD	:08
39B0	1A	16	FE	0E	30	8C	02	20	05	41	42	53	2B	00	34	10	:5E
39C0	17	FA	0F	32	62	10	27	00	15	17	FB	5C	32	62	CC	4D	:95
39D0	06	17	FA	52	32	62	CC	00	03	DD	1A	16	FA	E4	30	8C	:73
39E0	02	20	05	4E	4F	54	2B	00	34	10	17	F9	E5	32	62	10	:1D
39F0	27	00	15	17	FB	27	CC	4D	CD	34	06	17	FA	28	32	62	:62
Sum: 8F 87 C4 E8 31 20 92 88 97 9C 96 C9 BD 26 19 C3 :7E																	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3A00	CC	00	03	DD	1A	16	FA	B4	30	8C	02	20	05	41	44	52	:4A
3A10	2B	00	34	10	17	F9	B8	32	62	10	27	00	2C	17	FA	FD	:3C
3A20	DC	1A	B3	00	02	10	27	00	03	16	FA	50	CC	00	03	DD	:BF
3A30	1A	CC	00	C6	9E	04	E7	1E	CC	1F	B8	9E	04	ED	B4	DC	:E5
3A40	04	C3	00	02	DD	04	16	FA	79	30	8C	02	20	06	48	49	:A8
3A50	47	4B	2B	00	34	10	17	F9	79	32	62	10	27	00	20	17	:86
3A60	FA	B8	CC	1F	B9	9E	04	ED	B4	CC	00	4F	9E	04	E7	0E	:E2
3A70	DC	04	C3	00	03	DD	04	CC	00	03	DD	1A	16	FA	A3	30	:D0
3A80	8C	02	20	05	4C	4F	57	2B	00	34	10	17	F9	44	32	62	:F9
3A90	10	27	00	19	17	FA	B6	CC	00	4F	9E	04	E7	B4	DC	04	:EF
3AA0	C3	00	01	DD	04	CC	00	03	DD	1A	16	FA	15	30	8C	02	:4E
3AB0	20	05	53	45	5B	2B	00	34	10	17	F9	16	32	62	10	27	:72
3AC0	00	19	17	FA	5B	CC	00	1D	9E	04	E7	B4	DC	04	C3	00	:1B
3AD0	01	DD	04	CC	00	03	DD	1A	16	F9	E7	30	8C	02	20	05	:81
3AE0	53	47	4E	2B	00	34	10	17	FB	E8	32	62	10	27	00	4A	:60
3AF0	17	FA	2A	CC	00	03	DD	1A	CC	B3	00	9E	04	ED	B4	CC	:2F
Sum: F5 15 7B CE 85 F5 9F 49 3C 1E 61 6B 9F BD 68 44 :D0																	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3B00	00	27	9E	04	ED	02	CC	04	2A	9E	04	ED	0C	05	CC	0C	:E8
3B10	9E	04	ED	06	CC	FF	FF	9E	04	ED	08	CC	20	03	9E	04	:B7
3B20	ED	0A	CC	00	9E	04	ED	0C	CC	00	01	9E	04	E7	0E	0E	:8E
3B30	DC	04	C3	00	0F	DD	04	16	F9	B8	9E	16	4F	E6	B4	B3	:1A
3B40	00	2D	10	26	00	49	DC	16	C3	00	01	DD	16	17	FE	16	:80
3B50	DC	1A	B3	00	01	10	26	00	12	DC	24	53	43	C3	00	01	:1C
3B60	DD	24	DC	24	9E	04	ED	1E	16	00	21	CC	53	43	9E	04	:E9
3B70	ED	B4	CC	C3	00	9E	04	ED	02	CC	00	01	9E	04	E7	04	:C1
3B80	DC	04	C3	00	05	DD	04	CC	00	03	DD	1A	16	F9	33	30	:E1
3B90	8C	02	20	06	05	45	45	4B	2B	00	34	10	17	F8	33	32	:B9
3BA0	62	10	27	00	66	17	F9	75	DC	1A	B3	00	01	10	26	00	:34
3BB0	17	DC	04	C3	00	01	DD	04	CC	4F	9E	04	ED	1C	DC	34	:34
3BC0	24	9E	04	ED	1E	16	00	3D	CC	1A	B3	00	02	10	26	00	:D5
3BD0	18	CC	00	9E	04	ED	07	1E	CC	EC	B4	9E	04	ED	B4	DC	:54
3BE0	04	C3	00	02	DD	04	16	00	1C	CC	1F	01	9E	04	ED	B4	:DB
3BF0	CC	4F	E6	9E	04	ED	02	CC	00	B4	9E	04	E7	04	DC	04	:4F
Sum: FA 96 4D D7 BF BC E4 B3 B4 49 3E 3B 1B CD AC 22 :8C																	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3C00	C3	00	05	DD	04	CC	00	03	DD	1A	39	9E	16	4F	E6	B4	:15
3C10	B3	00	25	10	26	00	5B	DC	16	C3	00	01	DD	16	17	FD	:F6
3C20	45	DC	1A	B3	00	01	10	27	00	03	16	F6	4F	DC	04	B3	:B7
3C30	00	01	DD	04	CC	00	10	9E	04	E7	1E	CC	24	5B	49	DD	:BF
3C40	24	DC	24	B3	00	01	10	2C	00	EC	00	60	D3	24	9E	0C	:00
3C50	04	E7	1F	16	00	14	CC	00	EB	9E	04	E7	1F	DC	24	9E	:2E
3C60	04	E7	B4	DC	04	C3	00	01	DD	04	CC	00	06	DD	1A	16	:D3
3C70	FB	50	30	8C	02	20	05	52	4E	44	2B	00	34	10	17	F7	:89
3C80	51	32	62	10	27	00	15	17	FB	93	CC	4F	09	34	06	17	:48
3C90	F7	94	32	62	CC	00	03	DD	1A	16	FB	26	9E	16	4F	F6	:02
3CA0	B4	B3	00	2B	10	26	00	23	DC	16	C3	00	01	DD	16	17	:48
3CB0	FB	B7	9E	16	4F	E6	B4	B3	00	29	10	27	00	03	16	03	:E8
3CC0	6D	DC	16	C3	00	01	DD	16	16	F7	F7	9E	16	4F	E6	B4	:87
3CD0	B3	00	24	10	26	00	7B	CC	00	00	00	2E	CC	00	00	DD	:D5
3CE0	24	DC	2E	B3	00	09	10	2F	00	07	DC	2E	B3	00	07	DD	:71
3CF0	2E	DC	24	5B	49	5B	49	5B	49	5B	49	D3	2E	DD	24	DC	:90
Sum: B5 3B D6 D3 BD 42 82 26 57 F7 C1 C1 5A BB 55 5B :A5																	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3D00	16	C3	00	01	DD	16	9E	16	4F	E6	B4	B3	00	30	DD	2E	:F8
3D10	DC	2E	36	06	CC	00	09	BD	4D	A9	36	06	DC	2E	36	06	:50
3D20	CC	00	06	11	BD	4D	A1	A4	C0	EA	C0	36	06	DC	2E	36	:12
3D30	CC	00	00	BD	4D	A1	AA										

## リスト2 ランタイム・ルーチン ダンプ・リスト

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4D00 16 02 35 16 02 79 17 02 99 20 0C 35 10 20 02 BD :B0
4D10 EF A6 80 26 FA 6E B4 86 0D BD E5 B6 0A 20 E1 36 :F3
4D20 06 A6 43 3D 34 06 EC 41 3D E8 E4 E7 E4 A6 C4 E6 :BA
4D30 43 3D EB E0 A6 E0 1E 89 33 44 39 BE 00 00 34 10 :FA
4D40 34 10 34 10 4D 2A 04 8D 40 63 61 6C 60 2B 40 58 :23
4D50 49 2A F8 44 56 ED 64 EC C1 2A 04 8D 2C 63 61 6A :18
4D60 60 2B 1E 58 49 2A F8 44 56 A3 64 20 06 58 49 24 :F8
4D70 F8 E3 64 1C FE 2B 02 1A 01 69 63 69 62 6A 60 2A :2C
4D80 EC 62 6D 61 32 66 2A 05 43 53 C3 00 01 39 32 :94
4D90 66 CC 7F FF 39 4D 2B F1 79 A3 C1 2F 2C 0C 00 00 :16
4DA0 39 A3 C1 2E 24 CC 00 00 39 A3 C1 2D 1C 0C 00 00 :6D
4DB0 39 A3 C1 2C 14 CC 00 00 39 A3 C1 27 0C 0C 00 00 :45
4DC0 39 A3 C1 26 04 CC 00 00 39 CC 00 01 39 83 00 00 :55
4DD0 27 F7 CC 00 00 39 1F FF 2A 1F B9 4F 39 8D 15 16 :9B
4DE0 FF 47 BE 04 1D 1F FF 1E A6 80 B1 24 27 2B B1 2D :74
4DF0 27 EB 30 1F CC 00 00 34 06 A6 84 B2 30 2B 18 91 :07
Sum: 6D FD 3F 30 9F 6C AE 95 2D B2 5E FC 0F 01 0C BF :2D

```

```

4EA0 5F A6 C5 8B 30 17 FE 58 5C C1 06 26 F4 33 4A 39 :E5
4EB0 33 5E 8D 8C 20 E2 86 20 17 FF 45 50 26 FA 39 34 :93
4EC0 04 8D 04 35 04 1F 9B 1F 89 44 44 44 44 8D 04 1F :ED
4ED0 98 84 0F 88 30 B1 39 23 02 8B 07 16 FE 02 1F 9B :44
4EE0 20 F9 33 4C A6 45 E6 47 3D ED 42 EC 45 3D EB 42 :C7
4EF0 89 00 ED 41 A6 44 E6 47 3D E3 41 FD 41 A6 44 E6 :2D
Sum: 4A 81 AD 42 3C E6 90 34 05 57 25 69 AC 1D 01 C4 :AB

```

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4F00 46 3D EB 41 89 00 ED C4 39 34 06 EC 8C 1A BE 3D :B9
4F10 09 36 16 8D CC EC 42 33 48 C3 00 02 ED 8C 09 35 :D5
4F20 10 36 16 8D 8D 33 48 39 1C 2C 6A 49 16 FE 5A 20 :E3
4F30 7C CF 34 38 8C 00 00 00 20 02 20 13 34 31 8D 26 :B3
4F40 10 8E FC 82 8F 03 01 AF A1 A7 64 8D 2F 35 01 34 :1E
4F50 05 F6 FD 02 54 25 FA B7 FD 01 C6 00 F7 FD 00 CA :A6
4F60 04 F7 FD 00 35 B5 34 02 B6 FD 05 2B FB 1A 40 B6 :E2
4F70 80 87 FD 05 B6 FD 05 2A FB 35 82 7F FD 05 38 24 :B8
4F80 05 C6 12 8D 5D 8D DF CC 29 00 FD FE 82 3D 11 0D :A9
4F90 05 8D 02 35 85 16 80 B6 FC B3 FD 80 F7 FD 80 :83
4FA0 2D D9 34 34 C6 11 8D 7A 8D BC CC 04 02 FD FC E2 :0B
4FB0 C1 11 07 1D 11 3D 8D C1 BD AC EA FC 83 0A 2F EA :A7
4FC0 FC 84 27 12 5A 03 5A 27 0D 10 FE F7 89 06 70 27 :8A
4FD0 05 A7 80 5A 26 F7 C6 0D E7 84 3D B9 C6 12 8D 02 :8E
4FE0 05 85 8D 82 86 0C FD FC 82 20 90 41 F7 89 8D 05 :E1
4FF0 B6 FD 01 01 FF 27 06 FD 0F 7E 3F 1D 7E 94 8D :58
Sum: 62 F4 12 B2 15 37 47 EE FE AD 63 0D 84 BE 06 50 :4E

```

## リスト3 ファイル版メイン・プログラム

```

10 CLEAR100,&H6FFF
11 INPUT"HOT OR COLD":A$
12 IF A$="H" THEN 40
20 LOADM"RTF
30 LOADM"KF
40 INPUT"FILE NAME":A$
50 OPEN"I",#1,A$
55 IF PEEK(&H2DC)=0 THEN PRINT"*** NOT ASCII FILE ***":CLOSE #1:END
56 PRINT"FILE IS OPENED"
60 EXEC&H2000

```

## リスト4 ファイル版Kコンパイラ変更箇所

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4180 CC 10 27 DD 2C 39 34 48 BD 4D 06 B6 0D 10 9E 00 :12
4190 A7 A4 10 BE 00 00 EC 84 10 83 7F 0D 27 28 A6 80 :ED
41A0 81 0D 27 31 81 27 1E 80 30 2B F2 81 39 22 EE :68
41B0 34 02 1F 20 34 06 58 49 58 49 E3 E1 58 49 EB E0 :21
41C0 89 00 1F 02 20 D8 34 20 10 9E 00 A6 80 A7 A0 81 :92
41D0 0D 26 FB 35 20 10 9F 12 9E 00 9F 16 35 C8 12 DD :80
41E0 16 39 CC 00 33 DD 46 9E 46 EC 84 DD 46 39 BD D1 :AF
41F0 45 39 29 3D 24 44 4A 20 59 3A 36 29 3D 53 28 00 :5A
Sum: 19 5B 89 30 78 6F FC 23 F2 0D EC 2B 45 B5 EB 7D :A5

```

## リスト5 ファイル版ランタイム・ルーチン変更箇所

```

Add +0
4D09 39
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4FA0 2D D9 34 34 6F 4F 1F 8B C6 01 D7 8F BD DA 50 25 11 :0F
4FB0 0D C0 26 0D BE 04 3D A6 80 26 FC 86 0D A7 B2 20 :F3
4FC0 08 CC 7F 0D BE 04 3D ED 84 BE 04 3D 35 EF EF 27 :A9
Sum: 35 65 D9 B9 6B 27 05 59 05 8B BF 80 1C E6 96 58 :AB

```

## リスト6 リロケータ・プログラム・リスト

```

10 *PRINT "RELOCATER":POKE &FD0F,0
20 *PRINT "START ADR=":SA=INPUT
30 *PRINT "END ADR=":EA=INPUT
40 *PRINT "DEST ADR=":DA=INPUT
50 *WHILE SA=EA:DA=0:SA=0:DA=DA+1:SA=SA+1:WEND

```

## リスト7 エグゼキュータ・プログラム・リスト

```

10 *PRINT "EXECUTOR":POKE &FD0F,0
20 *PRINT "START ADR=":SA=INPUT
30 *CALL SA

```

## リスト8 リロケータ&amp;エグゼキュータ ダンプ・リスト

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
7C00 34 7F B6 90 00 B1 E1 27 0A 7F FD 0F 8D 3C 7D FD :5A
7C10 0F 35 FF BD 73 7F FD 05 CC 00 32 ED BD 01 25 BD :EF
7C20 34 BD 4A 7F FD 05 BD 19 BD 20 CC 00 96 ED BD 01 :C2
7C30 13 BD 28 BD 43 7F FD 05 BD 45 BD 4C 7F FD 05 35 :7A
7C40 FF 7F 90 00 B6 90 00 26 F8 39 8E 20 00 10 BE C0 :B7
7C50 00 A6 80 A7 A0 8C 42 00 26 F7 39 BD 3F BD 54 31 :6F
7C60 8D 00 C2 BE FC BD A6 A0 A7 80 26 FA 39 31 BD 00 :EA
7C70 C3 A6 A0 A7 80 26 FA 39 31 BD 00 BC BD F3 39 7F :3B
7C80 90 00 B6 90 00 27 F8 39 BD 12 BE FC B2 10 BE 02 :79
7C90 00 10 AF 81 10 BE 00 00 10 AF 84 39 B6 FD 05 2B :3D
7CA0 FB 1A 50 86 80 B7 FD 05 C6 AF 5A 26 FD B6 FD 05 :1E
7CB0 2A FB 39 BE FC B2 31 BD 00 B2 A6 A0 A7 80 81 FF :97
7CC0 26 FB 39 12 12 12 34 7F B6 90 00 B1 E1 27 0A 7F :94
7CD0 34 7F B6 90 00 B1 E1 27 0B 7F FD 0F 35 7F 7E C0 :07

```

```

7CE0 00 BD A5 7F FD 05 CC 00 64 ED BD 00 57 17 FF 6B :35
7CF0 17 FF 7A 7F FD 05 17 FF 4B BE 4F FE 1F 53 33 47 :36
Sum: 05 C1 95 CA 1D BE 68 B9 B3 ED 80 C4 9C 3B A3 52 :41
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
7D00 EF 84 BD C0 00 CC 00 AA ED BD 00 3B 17 FF 4C 17 :91
7D10 FF 66 7F FD 05 17 FF 67 BE 4F FE CE 9A BD EF 84 :A0
7D20 35 7F 7E 9A BD C3 DE AB AF CC DF 20 BD B2 AF C1 :F5
7D30 20 39 20 4F 46 46 00 AF 4E 4E 20 19 06 00 00 :30
7D40 02 02 00 00 00 CC 12 FF 00 00 00 00 00 00 00 :E1
Sum: 45 A4 DA 51 E1 B8 35 B8 79 F6 FD 26 81 44 EA 5C :37

```

## Kコンパイラ用

## グラフィック・サブルーチン

(Kコンパイラでグラフィック処理を!)

■コンピ

Kコンパイラの画面表示はPRINT文のみしか扱っていません。そこで、いままで発表されたグラフィック・サブルーチンを使いやすいように整理してみました。

以下にその文法を書きます(『FM活研』にある本来の文法は省略します)。

## ステートメントの文法

## \*CLS [消去範囲, 表示色, 背景色]

機能: 画面を消す。消去範囲は0が全画面, 1がスクロール画面, 2がページ1画面, 3がページ2画面。

例: CLS [0, 7, 0]

\*SCREEN [CX座標, CY座標<sup>1</sup>]

機能: 指定座標のキャラクタ・コードを返す。

例: A=SCREEN [18, 6]

## \*POINT [GX座標, GY座標, カラー]

機能: 指定座標の指定色のドットの有無を返す。

有=-1, 無=0

例: A=POINT [600, 172, 6]

\*LINE [GX座標, GY座標, GX座標, GY座標, 機能コード<sup>2</sup>, カラー, オプション・コード<sup>3</sup>]

機能: 線または四角を描く。

例: LINE [100, 20, 450, 60, 0, 6, 2]

## \*CLINE [CX座標, CY座標, CX座標, CY座標, 文字コード, カラー, オプション・コード]

機能: 文字による線, または四角を描く。

例: CLINE [0, 0, 10, 7, 'E', 6, 1]

## \*CONNECT [カラー, 機能コード, 指定座標数, GX座標, GY座標, GX座標, GY座標, ……]

機能: 指定座標を線で結ぶ。

例: CONNECT [4, 0, 3, 100, 60, 200, 30, 500, 180]

## \*SYMBOL [GX座標, GY座標, 縦倍率, 横倍率, カラー, 機能コード, 角度コード, 表字文字数, 文字列 (文字列の格納アドレス)]

機能: 文字を拡大表示する。

例: SYMBOL [0, 20, 5, 3, 4, 0, 0, 3, "I/O"]

## \*PSET [GX座標, GY座標, カラー]

機能: ドットを指定した色で表示する。

例: PSET [200, 50, 6]

## \*PRESET [GX座標, GY座標]

機能: 点を消す。

例: PRESET [300, 100]

## \*PAINT [GX座標, GY座標, カラー, 境界色数, 境界色1, 境界色2, 境界色3, ……]

機能: 指定座標から境界色の内部をぬる。

例: PAINT [100, 50, 6, 2, 3, 4]

## \*KANJI [GX座標, GY座標, 漢字コード]

機能: 漢字を表示する。

例: KANJI [320, 100, \$3E55]

## \*LOC [CX座標, CY座標]

機能: PRINTする位置を指定する。

例: LOC [10, 15]

## \*COLOR [カラー]

機能: 色を変える。

例: COLOR [2]

## \*WIDTH [横文字数, 縦行数]

機能: 画面の初期設定をする。

例: WIDTH [80, 25]

## \*CONSOLE [スクロール開始行, スクロール行数, 横文字数, 縦行数]

機能: 画面の初期設定をする。

例: CONSOLE [10, 5, 80, 25]

## \*INKEY [ ]

機能: おされたキーの内容を返す。

例: A=INKEY [ ]

注: A=GETのGET文はキーが押されるまで待ちますが, INKEY [ ]では待ちません。

## \*SETTIME [時, 分, 秒]

機能: 時間を設定する。

例: SETTIME [0, 0, 0]

## \*TIME [ ]

機能: 時間を返す。

例: A=TIME [ ]

注: 時=TH, 分=TM, 秒=TSである。

## \*HARDC1 [濃淡, 濃淡, …, 濃淡]

機能: ハード・コピーをとる (大)。

カラーコード0から7までの色のプリントされる  
ときの濃淡を0=無色, 1=淡い, 2=濃いで指  
定します。

## \*HARDC2 [ ]

機能：ハード・コピーをとる (小)。

例：HARDC 2 [ ]

リストはFM-8用なのでFM-7のときは、1110行～1130行の\$F2D8#を\$F17D#に変えてください。

本来の文法と上記の文法はFM-8、FM-7で完全コンパチです。

音の出ないFM-8のために、次のサブルーチンをおまけで載せておきます。

\* SOUND [音の長さ、音の高さ]

機能：スピーカーから音を出す。

音の長さは1から255までで指定します。長さは、何倍の長さかという指定ですから16は2の8倍の長さです。

音の高さは、指定する数が小さいほど、高くなります。つまり指定する高さは、音の周期みたいなものです。

表1に音階データを載せておきます。

例：SOUND [4, 35]

\* ANPORT [チャンネル、レンジ]

機能：指定したチャンネル、レンジからアナログ・データを読みとる。

レンジは0～2.5Vは'H' (\$48)、0～5/8Vは'L' (\$4C) です。

例：A=ANPORT [0, 'H']

4 : XOR  
5 : NOT  
0 : ノーマル  
1 : B  
2 : BF

注3：オプション・コード

表1

	C <sup>2</sup>	C <sup>3</sup>	C <sup>4</sup>	C <sup>5</sup>
C	231	114	55	26
C# D <sup>b</sup>	218	107	52	24
D	205	101	49	22
D# E <sup>b</sup>	194	95	46	21
E	183	89	43	19
F	172	84	40	18
F# G <sup>b</sup>	162	79	38	17
G	153	75	35	16
G# A <sup>b</sup>	144	70	33	15
A	136	66	31	14
A# B <sup>b</sup>	128	62	29	13
B	121	58	27	12

※有象無象より引用させていただきました。

## 最後に

これで、FM-8とFM-7は、BASICとKコンパイラ(一部のサブルーチンを除く)でコンパチになったわけです。

なお、ファイル版でコンパイルしたあとはCLOSEを忘れないように。

### 参考文献

“FM-8活用研究”，工学社

注1：CX, CY座標は      キャラクタ単位

                             ドット単位

注2：機能コード

0 : PSET

1 : PRESET

2 : OR

3 : AND

### プログラム・リスト

```

1000 '(* F-BASIC SUBROUTINE *)
1010 'CLS:HLT[]:CPOKE ADDR,$02,$1,$3,$2:SUB[]:RETURN
1020 'SCREEN:HLT[]:CPOKE ADDR,$06,$1,$2,$1,$2:SUB[]:HLT[]:DMY=PEEK($FCB4):POKE $
FCB0,$80:SUB[]:DMY=DMY:RETURN
1030 'POINT:HLT[]:CPOKE ADDR,$1B,$1#,$2#,$1#,$2#,$1,$3:SUB[]:HLT[]:DMY=PEEK($FCB4
):POKE $FCB0,$80:SUB[]:IF DMY>7F THEN DMY=-1 ELSE DMY=0:FI:DMY=DMY:RETURN
1040 'LINE:HLT[]:CPOKE ADDR,$15,$6,$5,$1#,$2#,$3#,$4#,$7:SUB[]:RETURN
1050 'CLINE:HLT[]:CPOKE ADDR,$20,$6,$5,$1#,$2#,$3#,$4#,$7:SUB[]:RETURN
1060 'CONNECT:HLT[]:CPOKE ADDR,$16,$1,$2,$3,$4#,$5#,$6#,$7#,$8#,$9#,$10#,$11#,$1
2#,$13#):SUB[]:RETURN
1070 'SYMBOL:DMY1=29:HLT[]:CPOKE ADDR,$19,$5,$6,$7,$3,$4,$1#,$2#,$8#:$FOR DMY=0 TO
$8-1:ADDR:DMY)=DMY1:DMY):NEXT:SUB[]:RETURN
1080 'PSET:HLT[]:CPOKE ADDR,$17,$1#,$2#,$3,0:SUB[]:RETURN
1090 'PRESET:HLT[]:CPOKE ADDR,$17,$1#,$2#,$1,1:SUB[]:RETURN
1100 'PAINT:HLT[]:CPOKE ADDR,$18,$1#,$2#,$3,$4,$5,$6,$7,$8,$9,$10:SUB[]:RETURN
1110 'KANJI:ADDR=$043D:CPOKE ADDR,$1600#,$0443#,$3#:$CODE $8E,$043D#,$BD,$F2D8#:$H
LT[]:ADDR=$FCB2:CPOKE ADDR,$1C,$1#,$2#,$1+15#,$2+15#,$7,0,$32:$DATA=$0443:ADDR=$FCB
E:$FOR DMY=0 TO 31:ADDR:DMY)=DATA:DMY):NEXT:SUB[]:RETURN
1120 'HARDC1:DMH=0:DML=0:HS[1,1]:HS[2,2]:HS[3,4]:HS[4,8]:HS[5,16]:HS[6,32]
:HS[7,64]:HS[8,128]:ADDR=$043D:CPOKE ADDR,$0F0F#,$0443#,$DMH,DML:$CODE $8E,$043D
#,$BD,$F2D8#:$RETURN:HS:IF $1=1 THEN DML=DML+2:FI:IF $1=2 THEN DMH=DMH+2:FI:RET
URN
1130 'HARDC2:ADDR=$043D:CPOKE ADDR,$0505#,$0441#:$CODE $8E,$043D#,$BD,$F2D8#:$RETU
RN
1140 'LOC:HLT[]:CPOKE ADDR,$3,$12,$1,$2:SUB[]:RETURN
1150 'COLOR:HLT[]:CPOKE ADDR,$3,$2,$1,$1:SUB[]:RETURN
1160 'WIDTH:HLT[]:CPOKE ADDR,$1,0,$1,$2,0,$2,0,$1,0:SUB[]:RETURN
1170 'CONSOLE:HLT[]:CPOKE ADDR,$1,0,$3,$4,$1,$2,0,$1,0:SUB[]:RETURN
1180 'INKEY:HLT[]:CPOKE ADDR,$29,0:SUB[]:HLT[]:DMY=PEEK($FCB3):POKE $FCB0,$80:$SU
B[]:DMY=DMY:RETURN
1190 'MOD:DMY=$1-$1/$2*$2:RETURN
1200 'SETTIME:HLT[]:CPOKE ADDR,$3D,$2,0,$1,$2,$3,0:SUB[]:RETURN
1210 'TIME:HLT[]:POKE $FCB2,$3E:SUB[]:HLT[]:TH=PEEK($FCB5):TM=PEEK($FCB6):TS=PEE
K($FCB7):POKE $FCB0,$80:SUB[]:DMY=TH*3600+TM*60+TS:RETURN
1220 'HLT:$CODE $B6,$FD05#,$2BF8#:$POKE $FD05,$80:$CODE $B6,$FD05#,$2AF8#:$ADDR=$FCB
2:$RETURN
1230 'SUB:POKE $FD05,0:$RETURN
1240 '
1250 '
1260 '(* FM-B ONLY *)
1270 '
1280 '
1290 'ANPORT:DMY=$1*256+$2:$CODE $A7,$8C10#,$E7,$8C0E#,$30,$8C0B#,$BD,$F2D8#,$4F,
$E6,$8C05#,$39,$0000#,$0000#,$00
1300 'SOUND:DMY=$1000/$2*$1:ADDR=$043D:CPOKE ADDR,$2#,$DMY:$CODE $7F,$0441#,$B6,$
0441#,$88B1#,$B7,$0441#,$B7,$FD03#,$FC,$043D#,$B3,$0001#,$26,$FBFC#,$043F#,$B3,$
0001#,$FD04#,$3F26#,$E2B6#,$01B7#,$FD03#,$39
1310 'BEEP:POKE $FD03,$41:RETURN:BEEP1:POKE $FD03,$81:RETURN:BEEP0:POKE $FD03,1:
RETURN

```



## Kコンパイラ用グラフィック・サブルーチン

■茶谷公之

Kコンパイラ用グラフィック・サブルーチンは非常に使いやすいものでしたが、発表されてないサブルーチンを下記に書きます（ゲーム中などではすでに発表されているものもあります）。

## 文 法

●CPUT1 [CX座標, CY座標, CX座標, CY座標, データの先頭アドレス, データのバイト数]

機能：対角線座標に示される枠内に文字を表示します。F-BASICのPUT@の形式1に対応します。ただし、データのバイト数は1~119の間でなければなりません。

例：CPUT1 [10, 10, 19, 14, \$6800, 50, 3]

●CPUT2 [CX座標, CY座標, CX座標, CY座標, データの先頭アドレス, データのバイト数, アトリビュート文字コード]

機能：対角線で示される枠内にアトリビュート文字と表示文字を転送します。F-BASICのPUT@の形式2に対応します。ただし、データのバイト数は1~119の間でなければなりません。

例：CPUT2 [10, 10, 19, 14, \$6800, 100]

●PUT2 [GX座標, GY座標, GX座標, GY座標, データの先頭アドレス, データのバイト数, カラーコード, ファンクション・コード]

機能：指定したわく内に指定色でパターンを書き込みます。F-BASICのPUT@の形式3に対応します。ただし、データのバイト数は1~114の間でなければなりません。

例：PUT1 [100, 100, 131, 115, \$6800, 64, 2, 0]

●PUT2 [GX, GY, GX, GY, データの先頭アドレス, データ数, ファンクション・コード]

機能：指定したわく内に3原色の画面データを書き込みます。ただし、データのバイト数は1~114の間でなければならず、ファンクション・コードは0と2-5で、1(PRESET)の指定はできません。

例：PUT2 [100, 100, 123, 107, \$6800, 72, 0]

●CHANGE [GX, GY, GX, GY, 旧カラーコード1, 新カラーコード2, 旧カラーコード2, 新カラーコード2, 旧カラーコード3, 新カラーコード3]

機能：指定した枠内の色を変更します。一応この命令では3色だけ変更できるようにしていますが、わずかな操作で変更させる色の数を1色~8色まで自由に定めることができます（FM-7システム仕様書参照）。

例：CHANGE [100, 100, 200, 150, 3, 5, 4, 7, 1, 2]

以下、FM-7のみにについて有効なサブルーチンです。

●SCREEN2 [アクティブVRAMコード, ディスプレイVRAMコード]

機能：アクティブ画面とディスプレイ画面の設定を行ないます。FM-7のSCREEN文とコンパチになっています。

例：SCREEN2 [4, 0]

●COLOR2 [パレットコード, カラーコード]

機能：カラーパレットの定義を行ないます。FM-7のCOLOR=の文とコンパチになっています。

例：COLOR2 [2, 4]

●SOUND [PSGのレジスタ番号, データ]

機能：FM-7のSOUND文とコンパチです。

例：SOUND [7, 0]

以上です。CPUT1, CPUT2, PUT1, PUT2についてはデータのバイト数に制限がありますが、少しの改良で制限バイト数以上でも可能になるので、各自必要に応じて変更してください。

また、GET, GCURSORについてもサブルーチン化することができますが、私自身がほとんど使わないコマンドなので、まだ作成していません。なお、サブルーチンはリスト1の通りです。

## ■参考文献

- 1) FM-7 ユーザーズ・マニュアル システム仕様
- 2) FM-7 F-BASIC 文法書
- 3) コンピ：Kコンパイラ用グラフィック・サブルーチン、I/O '83年4月号
- 4) FM-8活用研究、工学社

## グラフィック・サブルーチン・ソース・リスト

```

1000 ' (* GRAPHIC SUB ROUTINE *)
1010 '
1020 ' CPUT1:DE=%6-1:HLT[]:CPOKE ADDR,%07,%1,%2,%3,%4,%7,%6:DTD=%5:FOR WKI=0 TO Q
E:ADDR:WKI)=DTD:WKI):NEXT:SUB[]:RETURN
1030 '
1040 ' CPUT2:DE=%6-1:HLT[]:CPOKE ADDR,%09,%1,%2,%3,%4,%0,%6:DTD=%5:FOR WKI=0 TO DE
:ADDR:WKI)=DTD:WKI):NEXT:SUB[]:RETURN
1050 '
1060 ' PUT1:DE=%6-1:HLT[]:CPOKE ADDR,%1C,%1#,%2#,%3#,%4#,%7,%8,%6:DTD=%5:FOR WKI=
0 TO DE:ADDR:WKI)=DTD:WKI):NEXT:SUB[]:RETURN
1070 '
1080 ' PUT2:DE=%6-1:HLT[]:CPOKE ADDR,%1E,%1#,%2#,%3#,%4#,%0,%7,%6:DTD=%5:FOR WKI=0
TO DE:ADDR:WKI)=DTD:WKI):NEXT:SUB[]:RETURN
1090 '
1100 ' CHANGE:HLT[]:CPOKE ADDR,%1A,%1#,%2#,%3#,%4#,%3,%5,%6,%7,%8,%9,%10:SUB[]:RET
URN
1110 '
1120 ' SOUND:POKE $FD0E,%1:POKE $FD0D,%03:POKE $FD0D,%00:POKE $FD0E,%2:POKE $FD0D
,%02:POKE $FD0D,%00:RETURN
1130 '
1140 ' SCREEN2:DTD=ABS(%2-7))*16+ABS(%1-7):POKE $FD37,DTD:RETURN
1150 '
1160 ' COLOR2:POKE $FD3B+%1,2:RETURN
1170 '
1180 ' HLT:CODE $B6,$FD05#,$2BFB#:POKE $FD05,$B0:CODE $B6,$FD05#,$2AFB#:ADDR=$FCB
2:RETURN
1190 ' SUB:POKE $FD05,0:RETURN

```





# FM-7/8

# SUPER PATTERN EDITOR

## スーパーパターン・エディタ

市川浩志

ゲームを作ろうとすると、まず必要になるのがキャラクタ。最近では 640×200 ドットがあたりまえになって、キャラクタドット・パターンで作るのがほとんどですね。それも、だんだんアニメーションのようにリアルな動きになって、1つのキャラクタでもいろいろな方向のパターンが使われています。

こういう傾向はゲームをする方の側にとっては楽しいことですが、作る方の側になると大変なことで、それらのパターンを1つ1つ作るのはいくらパターン・エディタがあっても面倒なことです。

そこでお助けマン登場！このスーパーパターン・エディタはその面倒な作業をドンと引き受けてくれます。

写真1 パターンを作って

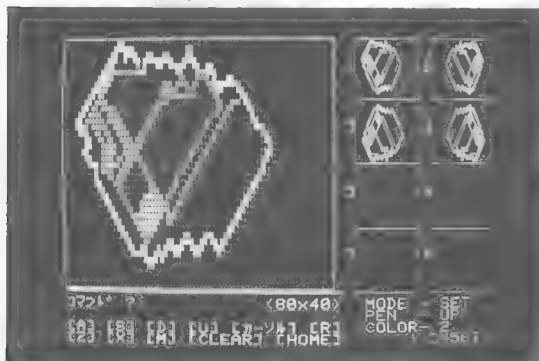
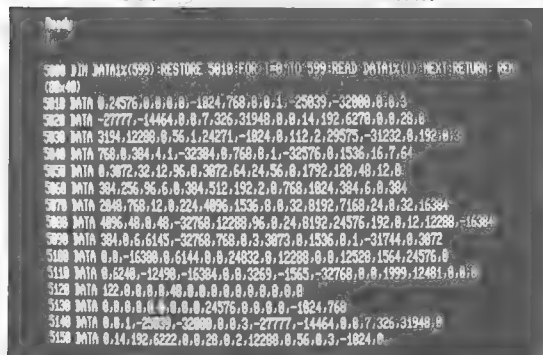


写真2 BASICのデータ文を作成



## 特 徴

- ①作成できるパターンの大きさは、最大80×40ドット。
- ②1つのキャラクタについて8つの方向のパターン・データを一度に作成する。
- ③データの形式はBASICのDATA文とマシン語のデータのどちらかを選べる。
- ④データを作成するフリーエリアが広い（16Kバイト）ので一度にたくさんのデータを作れる（ディスク・システムでも同じ）。
- ⑤FM-8、FM-7どちらでも使える。
- ⑥何といっても高速！

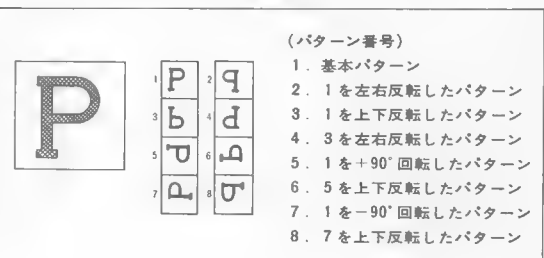
図1のように画面の右側には8つのわくがあって、左側の拡大パターンをキー操作で作っていきます。このとき右側のわくの中に8方向に向きを変えた実物大パターンがリアルタイムで出力されるので、それらを見ながら気に入るまでエディットできます。

このようなことをBASICでやると、ドットをセットしながらカーソルを移動するときなどかなり遅くなってしまのですが、Kコンパイラのオブジェクトなので操作性はバツグンです。そしてRコマンドによりそれらのデータを一度に作成できるので、データを作成する速さもBASICの100倍という超スピードですから、BASICで5分かかるものがわずかに3秒ででき上がってしまいます。

## 使 い 方

オブジェクト・プログラムをLOADM「ファイル名」、RまたはロードしたあとEXEC &H5000でプログラムがスタートします。CLEAR命令を実行する必要はありません。このことについてはあとでまた説明します。

図1 エディット画面



(パターン番号)

1. 基本パターン
2. 1を左右反転したパターン
3. 1を上下反転したパターン
4. 3を左右反転したパターン
5. 1を+90°回転したパターン
6. 5を上下反転したパターン
7. 1を-90°回転したパターン
8. 7を上下反転したパターン

## (OPTION選択)

画面をイニシャライズした後で“OPTION: +90° (Y/N)?”と聞いてきますが、これはパターン番号5～8のパターンを作成するかどうかを選択するものです。[N]を入力すると1～4までのパターンを作成します。これをモードIとします。[Y]を入力すると1～8までのパターンを作成します。これをモードIIとします。

なぜこんな選択をするのかというと、パターンを±90°回転するには横方向と縦方向の1ドットあたりの長さが等しくなければならないからです。640×200ドットのグラフィックでは横と縦のドットの長さの比率が1対2ですから、このまま±90°回転すると縦長のパターンになってしまいます。

これを避けるには横方向の2ドットを合わせて1ドットとすればいいわけで、実際モードIIではそのようにしています。つまり、モードIIでは実質的に320×200ドットのグラフィックとなってドットが少し荒くなります。

## (パターンの大きさ入力)

作成できるパターンの大きさはモードIで横が1～80ドット、縦が1～40ドットの範囲です。モードIIでは横が1～40ドットとなりますが、ドットの幅が2倍になっているので最大パターンの大きさはモードIと同じです。この制限を越えた数を入力するとBEEP音が鳴って再入力になります。

## (パターン・エディット)

図2のキーを操作してパターンを作ります。パターンが完成したら[R]を押してください。すると“コレディデスカ?(Y/N)”と聞いてきますからもう一度作成したパターンを点検してください。なにも修正する必要がなければ[Y]または[RETURN]キーを押します。

## (パターン番号入力)

次にどのパターンのデータを作成するのかを聞いてきますから、パターンの左側に付いている番号で入力してください。“1 2 3 5 …”のように連続して入力できます。また“1, 2, 3, 5, …”のように間にカンマを入れてもかまいません。パターン番号以外の数字や文字は無視されます。入力した数字を訂正したいときはスペース・キーを押してください。[RETURN]キーだけを押しとすべてのパターン(モードIのときは1～4まで)のデータを作成します。

図2 操作キー

[M] SET/RESETモード切り換え。	[1] ～ [7] ドットの色指定。
[U] ペン UP。	[+] カーソル位置にドットをセット。
[D] ペン DOWN。	[–] カーソル位置のドットをリセット。
[S] カーソル位置にドットをセット。	[R] パターン・データを作成する。
カーソルを1つ右へ。	
[A] カーソル位置にドットをセット。	[HOME] カーソルの色ですべてのカーソルを1つ左へ。
[X] カーソル位置のドットをリセット	[CLEAR] すべてのドットをクリア。
ト。カーソルを1つ右へ。	
[Z] カーソル位置のドットをリセット。	
ト。カーソルを1つ左へ。	
[←] [→] [↑] [↓] カーソル移動。ペンDOWNのときはドットをセット(またはリセット)しながらカーソル移動(セット・リセットの切り換えは[M]コマンドによる)。	

## (カラー、モノクロ選択)

パターン番号を入力した後は、カラーモードのデータを作るのか、モノクロ・モードのデータを作るのかを選択します。カラーモードはPUT@Aで表示するためのデータを作ります。モノクロ・モードはPUT@用の単色データです。カラーモードのデータ量はモノクロ・モードの3倍になります。

## (データ形式入力)

BASICで使うデータを作成する場合は[0]、マシン語またはKコンパイラ用のデータを作成する場合は[1]を入力してください。“Are You Sure(Y/N)?”と聞いてくるので押し間違えても訂正できます。

## (BASICデータ)

BASICのデータを選択すると“ヘンスウメイ?”と聞いてきますから、15文字以内でデータを読み込む配列の変数名を入力してください。最初が英字でないとエラーで再入力となります。入力した文字の訂正はスペース・キーを使います。15文字より長い変数名は入力できないようになっていきますから、いちいち数える必要はありません。15文字以内という制限があるのは、図3のように入力した変数名の最後に自動的にパターン番号を付け加えるようにしてあるからです。つまり15文字の変数名を入力した場合、作成されるデータ中の変数名は16文字になり、これはFBASICが変数名を区別できる最大の長さです。でもそんなに長い変数名を付ける人はまずいないでしょう。

図3 作成されるDATA文の例

入力した変数名	パターン番号
5000 DIM AKINA 3%(31): RESTORE 5010: FOR I= 0 TO...	
5010 DATA -1, 255, 13028, 3, -15, ……	
5020 DATA ……	

そのようなわけで、変数名の最後を見ればそのデータのパターン番号がわかりますからとても便利です。変数名の入力終了は[RETURN]キーです。

作成されるDATA文の行番号は5000から始まります。あらかじめBASICのプログラムやDATA文がメモリに入っている場合にはそれにアペンドします。そしてもしその最終行番号が5000より大きければ、アペンドされるデータの行番号はそれより10大きい数から始まります。

データの作成時間は長くても3秒以内です。小さなパターンならば一瞬ででき上がってしまいます。ビビビビ…とベルが鳴ってメモリの残りのバイト数が16進数で表示されると作成完了で、“Continue?(Y/N)”と聞いてきます。これ以上のデータを作らないときはもちろん[N]を入力してください。[Y]を入力すると次に“Same Character?(Y/N)”と聞いてきますから、また[Y]か[N]を入力してください。[Y]を入力すればいまの画面がそのまま残りますから、同じキャラクタで違うポーズのパターンを作るときは大変楽ですよ。

なお、残りバイト数がマイナスになって、“Memory Over!”と表示されたときにはもちろんそれ以上のデータは作成できません。そのときにはリストをとって、でき上がったDATA文の最後を点検してください。もし“Memory Over!”と入っていたらその部分のデータは使えません。

データを作るためのフリーエリアは約16Kバイトもありますが、それでも最大の大きさのパターン・データをカラー

モードで8つ作ると、16Kバイトをはるかにオーバーしてしまいます。ただし、たとえそのようなことをしても保護対策が万全ですからこのプログラム自体が破壊されるようなことはありません。試しに一度最大パターンのデータをカラーモードで作ってみてください。そのデータの大きさにきつとびっくりしますよ。

## (マシン語データ)

マシン語のデータを作る場合には“ADDRESS (\$1000-)?”と聞いてきますから、データの先頭アドレスを入力してください(16進数で入力するときには“\$”をつけるのを忘れないように)。\$1000より小さい数を入力したり、データの最終アドレスが\$4AF0を越えてしまう場合には再入力となります。ロードするときのオフセットで自由にアドレスを変えられますから、\$1000から作るのがよいと思います。

マシン語のデータはメモリ効率がよいので、最大の大きさのパターンでもカラーモードで8つ一度にデータを作ることができます。アドレスを入力すると、アッという間にデータが作成されてパターン番号とアドレスが出力されます。忘れずにそれをメモしてから何かキーを押してください。

このあとの手順はBASICデータの場合と同じです。

## プログラムについて

プログラムはKコンパイラで書かれています。BASICとはほぼ同じ文法ですが、ラベルがあるのでむしろBASICよりも解読しやすいのではないのでしょうか。

最後にあるCODE文は6809のマシン語サブルーチンでBASICデータの作成がその役目です。本当はこの部分もKコンパイラで書けるのですが、勉強を兼ねてマシン語に挑戦してみました。

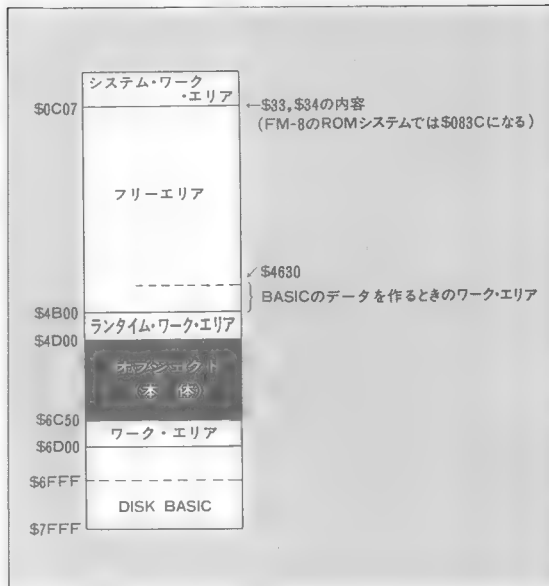
何をやっているかはアセンブル・リストを見てください。コメントがあるので解読はやさしいと思います。ただ、アセンブル・リストのアドレスは実際のアドレスではありませんから注意してください。このマシン語は完全にリロケータブルです。Kコンパイラに戻る直前にDレジスタにデータのエンド・アドレスを入れているので、作成したデータのエンド・アドレスを値とする関数になっています。

## コンパイルの方法

FM-8を使っている人は、ファイル版のKコンパイラでコンパイルしてください。カセットでもディスクでもコンパイルできます。

カセットを使う場合はソースをいくつかに分けてオン・メモリ版のコンパイラでデバッグするといいでしょう。FM-7のユーザーならばp.126の『拡張版Kコンパイラ』の記事を見てコンパイラを変更して、“フルRAMでコンパイルしてください。FM-8でもI/O'82年11月号の独立モニタを使えばフルRAMでコンパイルできるはずですが、まだ試していません。

図4 メモリ・マップ



CODE文のマシン語は当然1つでも間違えると暴走の危険があります。アセンブラを持っている人はアセンブル・リストを入力の方がいいかもしれません。その場合はKコンパイラのソース・リストの最初にDEF WORK=\$6A90を追加するだけでリンクできます。

## システム・スタック 《CLEAR命令》について

このプログラムのメモリ・マップを図4に示します。\$C07 (\$33, \$34の内容) から \$4B00までのフリーエリアの中にBASICまたはマシン語のデータを作成しますから、このエリアの中にシステム・スタックがあると暴走してしまいます。システム・スタックは必ず\$6D00より上のアドレスに置いてください。

一番安全なのはCLEAR, &H7FFF (ディスク・システムではCLEAR, &H6FFF) とすることです。つまりスイッチ・オン後、何もせずにこのプログラムをロードして走らせればよいのです。

コンパイル直後に走らせるときは、事前に必ずCLEAR, &H7FFF(またはCLEAR, &H6FFF)を実行してください。

## 最後に

このプログラムの使いごころはいかがですか。いままでのパターン・エディタよりはずっと便利になったと思います。

それとコマンドはすべて大文字で入力してください。

### 参考文献

- 1) “カラーパターン・エディタ”, I/O'82年3月号
- 2) FM-8活用研究

### リスト1 スーパーパターン・エディタ プログラム・リスト

```

1 ? (*-----*)
2 ? (* PATTERN EDITOR *)
3 ? (* Ver. 2.0 *)
4 ? (* '83.3 by IOLUM *)
5 ? (*-----*)
10 ?P=$6C50;Q=P+18;X=Q+18;Y=X+18;D=Y+18;HE=D+18;R=HE+20;CODE $1A40#

```

## リスト1 スーパーパターン・エディタ プログラム・リスト

```

20 'BEGIN; COL=7;CUX=0;CUY=0;SET=0;INIT[]
30 'LOC[0,21];?'OPTION: +90°, -90° (Y/N)?'
40 'L1; K=KEY[];IF K='Y' THEN B=2 ELSEIF K='N' THEN B=1 ELSE GOTO L1;FI;FI
50 'T=80/B;BEL[1];LOC[0,21];?'マトリックス サイズ' ?";SPC$(11)
60 'L2; LOC[2,22];?'ヨコ (1... ",T," ) ";MX=INPUT
70 ' IF MX<10R MX>T THENBEEP[];GOTO L2;FI
80 'L3; LOC[2,23];?'タテ (1...40) ";MY=INPUT
90 ' IF MY<10R MY>40THENBEEP[];GOTO L3;FI
100 'LINE[7,18,373,102,0,2]
110 'F(1)=440;P(2)=551+MX*B;P(3)=440;P(4)=551+MX*B;P(5)=439+MY*B;P(6)=551+MY*B;P
(7)=440;P(8)=552
120 'Q(1)=1;Q(2)=1;Q(3)=41+MY;Q(4)=41+MY;Q(5)=83;Q(6)=82+MX;Q(7)=123+MX;Q(8)=124
130 'D(1)= 1;D(2)=-1;D(3)= 1;D(4)=-1;D(5)=-1;D(6)=-1;D(7)= 1;D(8)=1
140 'X(1)= B;X(2)=-B;X(3)= B;X(4)=-B;X(5)=-B;X(6)=-B;X(7)= B;X(8)=B
150 'Y(1)= 1;Y(2)= 1;Y(3)=-1;Y(4)=-1;Y(5)= 1;Y(6)=-1;Y(7)=-1;Y(8)=1
160 'IF MX>50/B OR MY>22THEN U=4*B+1;V=4 ELSE U=7*B+1;V=7;FI
170 'LINE[0,0,MX*U+4,MY*V+2,5,1];TENTEN[];CURSET[];LINE[0,166,415,25,0,2]
180 'CMD0:
190 ' LOC[0,21];?'コマンド' ?";SPC$(12), "(",#2,MX,"x",#2,MY,")"
200 ' LOC[0,23];?'[A] [S] [D] [U] [カーソル] [R]",/, "[Z] [X] [M] [CLEAR] [HOME]"
210 'CMD; CL=7;K=KEY[]
220 ' IF K<56 AND K>48 THEN COL=K-48;LINE[592,184,45,7,COL,2];LOC[35,23];? COL;C
URSET[];GOTO CMD;FI
230 ' IF K=28 THEN Chiemi[];GOTO CXPL;FI
240 ' IF K=29 THEN Chiemi[];GOTO CXMI;FI
250 ' IF K=30 THEN Chiemi[];GOTO CYPL;FI
260 ' IF K=31 THEN Chiemi[];GOTO CYMI;FI
270 ' IF K='S' THEN DOTSET[];GOTO CXPL;FI
280 ' IF K='A' THEN DOTSET[];GOTO CXMI;FI
290 ' IF K='X' THEN DOTOFF[];GOTO CXPL;FI
300 ' IF K='Z' THEN DOTOFF[];GOTO CXMI;FI
310 ' IF K='U' THEN KF=1;CL=6;LOC[35,22];?"UP ";GOTO CMD;FI
320 ' IF K='D' THEN KF=0;CL=6;LOC[35,22];?"DOWN";GOTO CMD;FI
330 ' IF K='M' THEN CL=5;LOC[35,21];IF MF=0THEN MF=1;"RESET";ELSE MF=0;"SET ";
FI;GOTO CMD;FI
340 ' IF K='R' THEN GOTO MAKE;FI
350 ' IF K=12 THEN CS=0;FULL[];TENTEN[];CUX=0;CUY=0;CURSET[];GOTO CMD;FI
360 ' IF K=11 THEN CS=COL;FULL[];SET=4
370 ' I=2;WHILE I<MX*U+3;LINE[1,1,0,MY*V,COL,0];I=I+U;WEND
380 ' J=1;WHILE J<MY*V+2;LINE[2,J,MX*U,0,COL,0];J=J+V;WEND
390 ' SET=0;CUX=0;CUY=0;CURSET[];GOTO CMD;FI
400 ' IF K='+' THEN DOTSET[];GOTO CMD;FI
410 ' IF K='-' THEN DOTOFF[];GOTO CMD;FI
420 ' IF K=27 THEN END;FI
430 ' BEEP[];GOTO CMD0
440 '
450 ' (* Procedure *)
460 'FULL;
470 ' LINE[2,1,MX*U,MY*V,CS,2];FOR I=1TO4;L430[];NEXT
480 ' IF B=2THEN SWAP1[];FOR I=5TO8;L430[];NEXT;SWAP1[];FI;RETURN
490 'SWAP1; DMY=MX;MX=MY;MY=DMY;RETURN
500 'L430; LINE[P(I),Q(I),MX*X(I)-D(I), (MY-1)*Y(I),CS,2];RETURN
510 'Chiemi; IF KF=0 THENIF MF=1 THEN DOTOFF[] ELSE DOTSET[];FI;FI;RETURN
520 'TENTEN;
530 ' J=1;WHILE J<MY*V+2;I=2;WHILE I<MX*U+3;PSET[I,J];I=I+U;WEND;J=J+V;WEND
540 ' RETURN
550 'CXPL;
560 ' CUROFF[];CUX=CUX+1
570 ' IF CUX=MX THEN CUX=0;CUY=CUY+1;IF CUY=MY THEN CUY=0;FI;FI
580 ' CURSET[];GOTO CMD
590 'CXMI;
600 ' CUROFF[];CUX=CUX-1
610 ' IF CUX<0 THEN CUX=MX-1;CUY=CUY-1;IF CUY<0 THEN CUY=MY-1;FI;FI
620 ' CURSET[];GOTO CMD
630 'CYPL; CUROFF[];CUY=CUY-1;IF CUY<0 THEN CUY=MY-1;FI;CURSET[];GOTO CMD
640 'CYMI; CUROFF[];CUY=CUY+1;IF CUY=MY THEN CUY=0;FI;CURSET[];GOTO CMD
650 'CURSET; CX=CUX*U+3;CY=CUY*V+2;LINE[CX,CY+V-1,U-2,0,COL,0];RETURN
660 'CUROFF; LINE[CX,CY+V-1,U-2,0,0,0];RETURN
670 'DOTSET; CS=COL;GOTO Hidemi
680 'DOTOFF; CS=0
690 'Hidemi;
700 ' LINE[CX,CY,U-2,V-2,CS,2];FOR I=1TO4;L680[];NEXT
710 ' IF B=2THEN SWAP2[];FOR I=5TO8;L680[];NEXT;SWAP2[];FI;RETURN
720 'SWAP2; DMY=CUX;CUX=CUY;CUY=DMY;RETURN
730 'L680; LINE[CUX*X(I)+P(I),CUY*Y(I)+Q(I), (B=2)*D(I),0,CS,0];RETURN
740 ' (* -- Make Data -- *)
750 'MAKE;
760 'BEL[1];LINE[0,166,415,33,0,2];MX0=MX;MY0=MY
770 'LOC[3,21];?'コレサ イテスカ ? (Y/N)";K=KEY[]
780 'IF K<>'Y'AND K<>'13THEN GOTO CMD0 'FI;CLSC[];6=B*4
790 'Yumi; LOC[0,21];?'ヨタン No.?'";SPC$(28);FOR I=0TO8;R:I=0;NEXT;Junko[];CLSC[]
800 'IF HE:0=13THEN FOR J=1TO8;R:J=1;NEXT;L=6;GOTO Naoko;FI
810 'L=0;FOR J=0 TO I;K=HE:J;IF K<48 THENIF K<6+49 THENIF R:K-48=0THEN R:K-48
)=1;L=L+1;FI;FI;FI;NEXT
820 'IF L=0 THEN BEEP[];GOTO Yumi;FI
830 'Naoko; CL=6;LOC[3,21];?'[C] カラー モード";LOC[3,22];?'[M] モノクロ モード"

```

```

840 'Akina; SELE[];IF K='C'THENLOC[3,22] ELSEIF K='M'THEN LOC[3,21] ELSE GOTO Ak
ina;FI;FI
850 'SURE[];IF I='Y'OR I=13THEN BEL[1] ELSE GOTO Naoko;FI
860 'IF K='C'THEN MDE=1;T=MX*MY*B*3 ELSE MDE=2;T=MX*MY*B;FI
870 'C=T/16;IF T-T/16*16<>0 THEN C=C+1;FI
880 'You; CL=7;LOC[3,21];?'[1] マシコ" ",SPC$(15);LOC[3,22];?'[0] Basic DATA"
890 'IYO; SELE[];IF K='1'THENLOC[3,22] ELSEIF K='0'THEN LOC[3,21] ELSE GOTO IYO;
FI;FI
900 'SURE[];IF I='Y'OR I=13THEN CLS[]'ELSE GOTO You;FI
910 'Z=1;IF K='1'THEN MACHINE[]'ELSEIF K='0'THEN BASIC[];BEL[10] ;FI;FI
920 '?" Continue ? (Y/N) "; Mako[];IF K='N'THEN END ;FI;CL=6
930 'LOC[7,24];?"Same Character ? (Y/N) ";Mako[];IF K='N'THEN GOTO BEGIN;FI
940 'MX=MX0;MY=MY0;CLS[];INIT2[];GOTO CMD0
950 '
960 '(* Procedure *)
970 'Mako; BEL[1];REPEAT;K=KEY[];UNTIL K='Y'OR K='N';?CHR$(K);RETURN
980 'SELE; LOC[6,24];?"トチラニ シマスカ ? " ;K=KEY[];RETURN
990 'SURE; ?SPC$(14);LOC[6,24];?"Are You Sure ?(Y/N)";I=KEY[];RETURN
1000 'CLS; BEL[1];LINE[0,166,639,33,0,2];RETURN
1010 'PATRN;
1020 'IF N>4 AND Z THEN SWAP1[];Z=0;FI
1030 'X1=F(N);X2=MX*X(N)-D(N)+X1;IF X1<X2 THEN DMY=X1;X1=X2;X2=DMY;FI
1040 'Y1=G(N);Y2=(MY-1)*Y(N)+Y1;IF Y1<Y2 THEN DMY=Y1;Y1=Y2;Y2=DMY;FI
1050 'IF MDE=1 THEN PGet[$1D] ELSE PGet[$1B];FI;RETURN
1060 'Junko;
1070 'I=0;REPEAT;REPEAT;K=KEY[];UNTIL K=13 OR K>31;HE:I)=K
1080 'IF K='?' THENIF I>0THEN I=I-1;FI;PRT[];ELSEIF I<15THEN PRT[];I=I+1;FI;FI
1090 'UNTIL K=13 ;IF I<15 THEN I=I-1;FI;RETURN
1100 'PRT; LOC[11+I,21];?CHR$(K);RETURN
1110 'Lum; W=Z/10;DMY=W*256+Z-I-W*10+$3030;RETURN
1120 '(* -- Basic DATA -- *)
1130 'BASIC;
1140 'Seiko; LOC[1,21];?"ハズク メイ"?,SPC$(16);Junko[]
1150 'F=I;HE:I+1)=0;IF HE:0)<65 OR HE:0)>90THEN BEEP[];GOTO Seiko ;FI;BEL[1]
1160 'FOR N=1 TO G
1170 'IF R:N)=1THEN A=$4630;PATRN[];HE:F)=N+$30;I=Lum[MX*B];J=Lum[MY]
1180 'LAST=WORK[A,C,HE,I,J];FI
1190 'IF LAST>A THEN Yoshie[];BEL[6];?" Memory Over!";BEEP[];END;FI
1200 'NEXT
1210 'Yoshie; H=(A-LAST);?/,/, " [ ";IF H<0THEN?"-";FI;?"$",HEX$(ABS(H)), " "]
1220 'RETURN
1230 '(* - Machine DATA - *)
1240 'MACHINE;
1250 'LOC[5,23];?"16ビット ア [ $ ] ラ ツケテ !";I=0
1260 'Miyuki; LOC[0,21];?"ADDRESS($1000-) ? " ;AD1=INPUT
1270 'IF AD1<$1000 OR AD1>$4AF0-C*L*2 THEN BEEP[];GOTO Miyuki;FI;CLS[];LOC[0,21]
1280 'FOR N=1 TO G
1290 'IF R:N)=1THEN A=AD1;PATRN[];AD2=AD1+C*2-1;I=I+1
1300 '?"<"N,"> $" ,HEX$(AD1)," - $" ,HEX$(AD2);IF I<>8 THEN ?" " ;FI
1310 'AD1=AD2+1;FI;NEXT;BEL[10];IF I>4THEN KEY[];CLS[];FI;LOC[5,23];RETURN
1320 '
1330 'INIT;
1340 'CONF80,11;CL=7;LINE[0,163,417,1,4,2]
1350 'FOR I=0TO123 STEP 41;LINE[433,I,94,41,6,1];LINE[545,I,94,41,6,1];NEXT
1360 'J=2;FOR I=1TO4;LOC[53,J];? I*2-1;J=J+5;NEXT
1370 'J=2;FOR I=1TO4;LOC[67,J];? I*2 ;J=J+5;NEXT;CS=0;J=5
1380 'SYMBOL[10,20,"SUPER"]
1390 'SYMBOL[ 7,18,"SUPER"];J=7
1400 'SYMBOL[50,50,"PATTERN"]
1410 'SYMBOL[47,48,"PATTERN"];J=6
1420 'SYMBOL[90,80,"EDITOR"]
1430 'SYMBOL[87,78,"EDITOR"];CONF40,0]
1440 'INIT2; MF=1;KF=1
1450 'CL=5;LOC[28,21];?"MODE - RESET"
1460 'CL=6;LOC[28,22];?"PEN - UP"
1470 'CL=7;LOC[28,23];?"COLOR- ",COL
1480 'LINE[592,184,45,7,COL,2]
1490 'FOR I=1TO7;CL=I;LOC[31+I,24];? I;NEXT
1500 'RETURN
1510 '
1520 '(* PROCEDURES *)
1530 'BEEP; POKE $FD03,65;RETURN
1540 'BEL; FOR IS=1TOZ1;WAIT[];POKE $FD03,$B1;WAIT[];POKE $FD03,1;NEXT;RETURN
1550 'WAIT; FOR WJ=1TO 2000;NEXT;RETURN
1560 'LOC; HLT[];CPOKE AD,3,5,18,%1,%2,17,CL+$B0;SUB[];RETURN
1570 'KEY; REPEAT;W=GET;UNTIL W<>0;RETURN
1580 'CON; HLT[];CPOKE AD,1,0,%1,25,0,%2,0;SUB[];RETURN
1590 'PGet; HLT[];CPOKE AD,%1,X1#,Y1#,X2#,Y2#,7,1,2,3,4,5,6,7;SUB[];CNT=0
1600 'CODE $B618#,$4A,$26FD#
1610 'CYCL; HLT[];AD=$FCB4;WI=PEEK($FCB3)-1;FOR IS=0 TO WI ;A:CNT)=AD:IS);CNT=
CNT+1;NEXT
1620 'IF PEEK($FCB1)=$B0THEN POKE $FCB2,100;SUB[];GOTO CYCL ELSE POKE $FCB0,$B
0;SUB[];FI ;RETURN
1630 'LINE; HLT[];CPOKE AD,21,%5,SET,%1#,%2#,(%1+%3)#,(%2+%4)#,%6;SUB[];RETURN
1640 'PSET; HLT[];CPOKE AD,$17,1,%1#,%2#,%COL,0;SUB[];RETURN

```

## リスト1 スーパーパターン・エディタ プログラム・リスト

```

1650 *HLT; CODE $B6,$FD,5,$2BF#;POKE $FD05,$B0;CODE $B6,$FD,5,$2AFB#;AD=$FCB2;
RETURN
1660 *SUB; POKE $FD05,0;RETURN
1670 *SYMBOL; CS=CS+1;HLTC;CPOKE AD,$19,CS,0,0,6,3,%1#,%2#,J;QE=%3;FOR I=0TO J-
1;AD:I)=QE:I);NEXT;SUBC;RETURN
1680 *
1700 *WORK;
1710 * CODE $EC,$62,$ED,$8D,$01,$74,$EC,$64,$ED,$8D,$01,$66,$EC,$66,$ED,$8D
1720 * CODE $01,$6A,$EC,$68,$ED,$8D,$01,$66,$EC,$6A,$ED,$8D,$01,$62,$BE,$00
1730 * CODE $35,$30,$1E,$AF,$8D,$01,$51,$BC,$00,$33,$27,$16,$10,$BE,$00,$33
1740 * CODE $10,$AE,$A4,$6D,$B4,$26,$F9,$EC,$22,$C3,$00,$0A,$10,$B3,$13,$92
1750 * CODE $2C,$03,$CC,$13,$88,$ED,$8D,$01,$2D,$34,$40,$30,$02,$10,$AE,$8D
1760 * CODE $01,$24,$10,$AF,$81,$31,$2A,$10,$AF,$8D,$01,$1A,$CC,$84,$20,$ED
1770 * CODE $81,$8D,$4A,$10,$8E,$00,$20,$33,$8D,$01,$17,$EC,$8D,$01,$03,$83
1780 * CODE $00,$01,$ED,$44,$ED,$C8,$1B,$EC,$8D,$00,$FB,$ED,$4C,$8D,$3B,$8D
1790 * CODE $2C,$10,$8E,$00,$12,$33,$8D,$01,$19,$EC,$8D,$00,$F1,$ED,$4C,$EC
1800 * CODE $8D,$00,$ED,$ED,$4F,$8D,$23,$17,$00,$C5,$8D,$27,$6F,$80,$6F,$80
1810 * CODE $BF,$00,$35,$BF,$00,$3B,$BF,$00,$3D,$1F,$10,$35,$C0,$EE,$8D,$00
1820 * CODE $C8,$A6,$C0,$A7,$80,$6D,$C4,$26,$F8,$39,$A6,$C0,$A7,$80,$31,$3F
1830 * CODE $26,$F8,$39,$EE,$8D,$00,$B3,$10,$AE,$8D,$00,$AA,$30,$02,$10,$AF
1840 * CODE $81,$31,$2A,$34,$20,$CC,$83,$20,$ED,$81,$C6,$10,$8C,$47,$00,$25
1850 * CODE $0E,$10,$8E,$00,$0C,$33,$8D,$00,$CB,$8D,$CF,$35,$20,$70,$34
1860 * CODE $04,$37,$06,$31,$8C,$72,$6F,$8C,$7C,$10,$83,$00,$00,$27,$39,$2A
1870 * CODE $0C,$34,$02,$86,$2D,$A7,$80,$35,$02,$40,$50,$B2,$00,$6F,$8C,$64
1880 * CODE $10,$A3,$A4,$25,$07,$A3,$A4,$6C,$8C,$5A,$20,$F4,$34,$02,$A6,$8C
1890 * CODE $53,$4D,$26,$05,$6D,$8C,$4E,$27,$07,$8B,$30,$A7,$80,$6C,$8C,$45
1900 * CODE $35,$02,$31,$22,$6D,$21,$26,$D5,$CB,$30,$E7,$80,$EC,$8C,$33,$83
1910 * CODE $00,$01,$27,$15,$ED,$8C,$2B,$35,$04,$5A,$26,$07,$8D,$11,$35,$20
1920 * CODE $16,$FF,$79,$86,$2C,$A7,$80,$20,$96,$35,$24,$10,$AF,$8C,$17,$6F
1930 * CODE $80,$AF,$9C,$14,$AF,$8C,$11,$39,$27,$10,$03,$EB,$00,$64,$00,$0A
1940 * CODE $00,$00,$00,$20,$00,$00,$13,$8B,$31,$00,$4B,$00,$6C,$10,$00,$00
1950 * CODE $00,$00,$25,$28,$FE,$02,$00,$00,$29,$3A,$8A,$20,$FE,$F2,$00,$00
1960 * CODE $3A,$B1,$20,$49,$E6,$FE,$01,$00,$20,$CD,$20,$FE,$02,$00,$00,$3A
1970 * CODE $85,$20,$25,$28,$49,$29,$3A,$82,$3A,$8B,$3A,$20,$8C,$28,$20,$20
1980 * CODE $7B,$20,$20,$29,$4D,$65,$6D,$6F,$72,$79,$20,$4F,$76,$65,$72,$21

```

## リスト2 マシン語部アセンブル・リスト

```

0001 0000 :
0002 0000 : * Make Basic Data *
0003 0000 : * 'K'subroutine *
0004 0000 : * '83.3 by IOLUM *
0005 0000 :
0006 0000 :
0007 6A90 : ORG $6A90
0008 6A90 EC62 LDD 2,S ; Get %1 < Address of A(0) >
0009 6A92 ED8D0174 STD ADRES,PC
0010 6A96 EC64 LDD 4,S ; Get %2 (C)
0011 6A98 ED8D0166 STD COUNT,PC
0012 6A9C EC66 LDD 6,S ; Get %3 (HE)
0013 6A9E ED8D016A STD HE,PC
0014 6AA2 EC68 LDD 8,S ; Get %4 (MX)
0015 6AA4 ED8D0166 STD MX,PC
0016 6AAB EC6A LDD 10,S ; Get %5 (MY)
0017 6AAA ED8D0162 STD MY,PC
0018 6AAE BE0035 LDX #0035 ; Get End of Text
0019 6AB1 301E LEAX -2,X
0020 6AB3 AFB0D151 STX PQ,PC
0021 6AB7 BC0033 CMPX #0033 ; Text exist ?
0022 6ABA 2716 BEQ GL2 ; IF No THEN GL2
0023 6ABC 10BE0033 LDY #0033 ; Get Last Line Number of Text
t
0024 6AC0 10AEA4 GL1: LDY ,Y
0025 6AC3 6DB4 TST [,Y]
0026 6AC5 26F9 BNE GL1
0027 6AC7 EC22 LDD 2,Y
0028 6AC9 C3000A ADDD #10
0029 6ACC 10831392 CMPD #5010
0030 6AD0 2C03 BGE GL3
0031 6AD2 CC1388 GL2: LDD #5000
0032 6AD5 ED8D012D GL3: STD L.N,PC
0033 6AD9 3440 PSHS U
0034 6ADB :
0035 6ADB 3002 LEAX 2,X ; Set Link Pointer Area
0036 6ADD 10AEBD0124 LDY L.N,PC ; Set Line Number
0037 6AE2 10AFB1 STY ,X++
0038 6AE5 312A LEAY 10,Y
0039 6AE7 10AFBD011A STY L.N,PC
0040 6AEC CCB420 LDD #B420 ; Set "DIM "
0041 6AEF ED81 STD ,X++
0042 6AF1 BD4A BSR HENSU ; Set "ヘンズウ"
0043 6AF3 10BE0020 LDY #20 ; Set "%( ):RESTORE ____:"

```

## リスト2 マシン語部アセンブル・リスト

```

0044 6AF7 338D0117      LEAU D1,PC      ; "FOR I=0 TO __:READ __
0045 6AFB EC8D0103      LDD COUNT,PC   ; Set C in Data
0046 6AFF 830001         SUBD #1
0047 6B02 ED44          STD 4,U
0048 6B04 EDC81B        STD 27,U
0049 6B07 EC8D00FB      LDD L.N,PC      ; Set L.No. in Data
0050 6B0B ED4C          STD 12,U
0051 6B0D 8D3B          BSR TRANS
0052 6B0F 8D2C          BSR HENSU
0053 6B11 10BE0012      LDY ##12      ; Set "ヘンズウ"
                                ; Set "%(I):NEXT:RETURN: REM(
...X...
0054 6B15 338D0119      LEAU D2,PC
0055 6B19 EC8D00F1      LDD MX,PC      ; Set MX in Data
0056 6B1D ED4C          STD 12,U
0057 6B1F EC8D00ED      LDD MY,PC      ; Set MY in Data
0058 6B23 ED4F          STD 15,U
0059 6B25 8D23          BSR TRANS
0060 6B27 1700C5        LBSR POSET
0061 6B2A 8D27          BSR D.SET      ; Set "0000 DATA ..... "
0062 6B2C
0063 6B2C 6F80          CLR ,X+      ; Set System Pointer
0064 6B2E 6F80          CLR ,X+
0065 6B30 BF0035        STX #0035
0066 6B33 BF003B        STX #003B
0067 6B36 BF003D        STX #003D
0068 6B39
0069 6B39 1F10          TFR X,D
0070 6B3B 35C0          PULS U,PC      ; Return to K-Compiler
0071 6B3D
0072 6B3D
0073 6B3D
0074 6B3D EE8D00CB      HENSU: LDU HE,PC
0075 6B41 A6C0          HLOOP: LDA ,U+
0076 6B43 A780          STA ,X+
0077 6B45 6DC4          TST ,U
0078 6B47 26F8          BNE HLOOP
0079 6B49 39           RTS
0080 6B4A
0081 6B4A A6C0          TRANS: LDA ,U+
0082 6B4C A780          STA ,X+
0083 6B4E 313F          LEAY -1,Y
0084 6B50 26F8          BNE TRANS
0085 6B52 39           RTS
0086 6B53
0087 6B53      ; GET A(I) 0 TO COUNT
0088 6B53
0089 6B53      D.SET= *
0090 6B53 EE8D00B3      LDU ADRES,PC
0091 6B57 10AE8D00AA      LDY L.N,PC
0092 6B5C 3002          HEAD: LEAX 2,X      ; Set POINTER Area
0093 6B5E 10AF81          STY ,X++      ; LINE NO. SET
0094 6B61 312A          LEAY 10,Y      ; L.N=L.N+10
0095 6B63 3420          PSHS Y
0096 6B65 CC8320        LDD ##8320      ; "DATA " SET
0097 6B68 EDB1          STD ,X++      ; [ "5000 DATA " ]
0098 6B6A C610          LDB #16
0099 6B6C 8C4700        CMPX ##4700      ; Memory Over ?
0100 6B6F 250E          BCS A(I)      ; IF No THEN A(I)
0101 6B71 10BE000C      LDY #12
0102 6B75 338D00CB      LEAU OVER,PC
0103 6B79 8DCF          BSR TRANS
0104 6B7B 3520          PULS Y
0105 6B7D 2070          BRA POSET
0106 6B7F 3404          A(I): PSHS B
0107 6B81 3706          PULU A,B      ; Get A(I)
0108 6B83 318C72          LEAY <DATA,PC
0109 6B86 6F8C7C          CLR <FLAG,PC
0110 6B89 10B30000        CMPD ##0000      ; A(I)=0 ?
0111 6B8D 2739          BEQ L4      ; IF A(I)=0 THEN L4
0112 6B8F 2A0C          BPL LOOP      ; IF A(I)>0 THEN LOOP
0113 6B91 3402          PSHS A
0114 6B93 862D          LDA #'-'
0115 6B95 3502          PULS A
0116 6B97 40           NEGB
0117 6B98 50           NEGB
0118 6B99 8200          SBCA #0
0119 6B9B 6F8C66          LOOP: CLR <IY0,PC
0120 6B9E 10A3A4          L1:  CMPD ,Y
0121 6BA1 2509          BCS L2      ; IF A(I)<10000 THEN L2
0122 6BA3 A3A4          SUBD ,Y
0123 6BA5 6C8C5C          INC <IY0,PC
0124 6BA8 20F6          BRA L1
0125 6BAA 3402          L2:  PSHS A
0126 6BAC A68C55          LDA <IY0,PC
0127 6BAF 4D           TSTA

```

## リスト2 マシン語部アセンブル・リスト

```

0128 6BB0 2607      BNE L21
0129 6BB2 6B8C50    TST <FLAG,PC
0130 6BB5 2709      BEQ L25
0131 6BB7 8B30      L21: ADDA #$30
0132 6BB9 A780      STA ,X+
0133 6BBB 6C8C47    INC <FLAG,PC
0134 6BBE 3502      L25: PULS A
0135 6BC0 3122      L3: LEAY 2,Y
0136 6BC2 6D21      TST 1,Y
0137 6BC4 26D7      BNE LOOP      ; IF DATA>0 THEN LOOP
0138 6BC6 CB30      L4: ADDB #$30
0139 6BC8 E780      STB ,X+
0140 6BCA EC8C35    LDD <COUNT,PC
0141 6BCD 830001    SUBD #1
0142 6BD0 2717      BEQ RET
0143 6BD2 ED8C2D    STD <COUNT,PC
0144 6BD5 3504      PULS B
0145 6BD7 5A        DECB
0146 6BD8 2609      BNE L5
0147 6BDA BD13      BSR POSET
0148 6BDC 3520      PULS Y
0149 6BDE 16FF7B    LBRA HEAD
0150 6BE1 862C      L5: LDA #$2C      ; ", " SET
0151 6BE3 A780      STA ,X+
0152 6BE5 2098      BRA A(I)
0153 6BE7 3524      RET: PULS B,Y
0154 6BE9 10AFBC19 STY <L.N,PC
0155 6BED 6F80      POSET: CLR ,X+
0156 6BEF AF9C16    STX [<PO,PC]
0157 6BF2 AF8C13    STX <PO,PC
0158 6BF5 39        RTS
0159 6BF6          ;
0160 6BF6          ; Data Area
0161 6BF6          ;
0162 6BF6 2710      DATA: FDB 10000
0163 6BF8 03EB      FDB 1000
0164 6BFA 0064      FDB 100
0165 6BFC 000A      FDB 10
0166 6BFE 0000      FDB 0
0167 6C00 0020      COUNT: FDB $0020
0168 6C02 00        IY0: FCB 0
0169 6C03 00        FLAG: FCB 0
0170 6C04 1388      L.N: FDB 5000
0171 6C06 3100      PO: FDB $3100
0172 6C08 4800      ADRES: FDB $4800
0173 6C0A 6C10      HE: FDB $6C10      ; Address of HE:0)
0174 6C0C          MX: RMB 2
0175 6C0E          MY: RMB 2
0176 6C10          ;
0177 6C10 2528      D1: FDB $2528      ; %( ):RESTORE __:
0178 6C12 FE02      FDB $FE02      ; FOR I=0 TO __:READ __
0179 6C14 0000      FDB $0000
0180 6C16 293A      FDB $293A
0181 6C18 8A20      FDB $8A20
0182 6C1A FEF2      FDB $FEF2
0183 6C1C 0000      FDB $0000
0184 6C1E 3AB1      FDB $3AB1
0185 6C20 2049      FDB $2049
0186 6C22 E6FE      FDB $E6FE
0187 6C24 0100      FDB $0100
0188 6C26 20CD      FDB $20CD
0189 6C28 20FE      FDB $20FE
0190 6C2A 0200      FDB $0200
0191 6C2C 003A      FDB $003A
0192 6C2E 8520      FDB $8520
0193 6C30          ;
0194 6C30 2528      D2: FDB $2528      ; %( ):NEXT:RETURN: ( __x__ )
0195 6C32 4929      FDB $4929
0196 6C34 3AB2      FDB $3AB2
0197 6C36 3AB8      FDB $3AB8
0198 6C38 3A20      FDB $3A20
0199 6C3A 8C28      FDB $8C28
0200 6C3C 2020      FDB $2020
0201 6C3E 7820      FDB $7820
0202 6C40 2029      FDB $2029
0203 6C42          ;
0204 6C42 4D65D6F72 OVER: FCC /Memory Over!//
79204F7665
7221
0205 6C4E          ;
0206 6C4E          ; END

```





FM-7/8

Kコンパイラによる

# 日本語ワード・プロセッサ

■TTC.T.S

## I 概要

このプログラムは次のような項目を実現しようと作成しました。

- ①誰にでも使えるよう操作を簡単にする。
- ②カセット・ベースでも使えるようにする。
- ③待ち時間が気にならない程度に高速化する。
- ④漢字プリンタを使わないで、実用的なものとする。
- ⑤ローマ字入力を可能にする。

①を満たすため上下左右スクロールを可能にしたり、③を満たすためにプログラム本体をKコンパイラで作成したほか、サブシステムへ漢字表示やスクロール・プログラムを転送したりしています。④についてはMP80-II III用になっています。

## II プログラムの入力

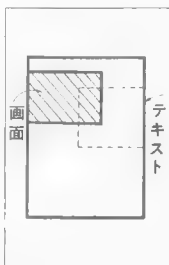
まず、リスト1(BASIC)、リスト2(マシン語)を入力し、それぞれ“JWP”、“JWPM”というファイル名で、テープ順にセーブします。次にリスト3(BASIC)、リスト4(マシン語)を入力して、“PRT”、“PRTM”というファイル名で同様にセーブします。

なお、このプログラムをテープ・ベースでしか使わない場合はリスト1の1220行の“ASZ=&H6FFF”を“ASZ=&H7FFF”に変更して入力してください。

## III 操作方法

### 1 起動

図1



- ①BASIC部分(ファイル名“JWP”)をロードしてRunする。
- ②“HOT OR COLD(H/C)”と表示するので、マシン語部分をロードする場合は“C”を入力する。
- ③“DATA LOAD(Y/N)”と表示するので、データをロードするときは“Y”を入力し続いてファイル名を、できない場合は“N”を入力し横文字数(40~60)を入力する。

### 2 基本的な使用

画面とテキストの関係は図1のようになっています。カーソルが画面の外へ出たら、画面の位置が変わってカーソルがまた画面の中に入ります。なお[RETURN]キーの入力に関係なくテキストが変化するので注意してください。画面の説明を図2に示します。

### 3 各キーの説明

- ①英数字、英記号、カナ、カナ記号、グラフィックなどのキーを押すと、そのまま入力されます。グラフィックの場合、キー配置は図3のとおり、ただし、これはFM-8の場合で、FM-7では若干異なります。
- ②カーソル・キー([↑][↓][←][→])で、カーソルを上下左右へ移動します。[SHIFT][←][SHIFT][→]でカーソルを左右へ5文字ごと、[SHIFT][↑][SHIFT][↓]で画面の左端へ移動した後、上下へ3文字ごと移動。

図2

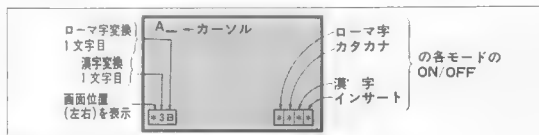


図3 GRAPHモードのキー配置(FM-8)

(STP)	(PF6)	(PF7)	(PF8)	(PF9)	(PF0)	(EL)	(CL)	(INS)	(↑)	(DEL)
	(PF1)	(PF2)	(PF3)	(PF4)	(PF5)	(DP)	(HM)	(←)	(↓)	(→)
{ESC}	T	~	...	*	'	[ ]	¢	∞	℃	{BS}
{CAP}	{	}	あ	い	う	え	お	か	き	{RETURN}
{CTRL}	「	」	あ	い	う	え	お	か	き	{カナ}
{SHIFT}		..	8	9	0	1	2	3	4	{SHIFT}
{G}	{TAB}	{								{RET}
										スペース

- ③ ファンクション・キーと **CTRL** - **X** を使って、すべてのファンクション・キーに38文字までの文字列を定義し呼び出すことができます。

#### ■定義のしかた

- (a) 定義する文字列の先頭へカーソルを移動。  
 (b) **SHIFT** - **X** を押す。  
 (c) ファンクション・キーの番号を入力。  
 (d) **DUP** を押してカーソルを定義する文字列の終わりの次の文字位置へ移動。  
 (e) **DUP** 以外のキーを押す。

もし38文字以上になったらブザーが鳴るので **DUP** 以外のキーを押す。

#### ■ファンクション・キーの内容の表示

- (a) **CTRL** - **X** を押す。  
 (b) **RETURN** キーを押す。  
 (c) 適当なキーを押して、テキストの表示にする。

#### ■使用（呼び出し）

文字列の定義してあるファンクション・キーを押せば、テキストに入力される。

- ④ **INS**、インサート・モードをON/OFFする。ON/OFFの状態は右下の四角の右端へ表示。  
 ⑤ **DEL**、**BS**、**EL**、**TAB** はF BASICと同じ。  
 ⑥ **HOME**、カーソルが画面左上端にあれば、カーソルをテキスト左上端へ、そうでなければ画面左上端へ移動。  
 ⑦ **CLEAR** (FM 7では **CLS**)、確認するので間違いがなければ“1”を入力します。  
 ⑧ **RETURN**、カーソルを次の行の先頭へ移動。  
 ⑨ **DUP**、上の行の文字を1文字ずつコピー。  
 ⑩ **ESC**、漢字モードをON/OFFする。状態は右下の四角の右から2番めに表示。  
 ⑪ **CTRL** - **B**、ローマ字モードをON/OFFする。状態は右下の四角の左端へ表示。  
 ⑫ **CTRL** - **N**、ひらがなモード/カタカナ・モードを選択する。状態は左から2番めに表示。  
 ⑬ **CTRL** - **A**、**CTRL** - **D**、カーソル位置の文字コードをインクリメント、デクリメントします。

<例>

**CTRL** **A** のとき“ひ”は“び”になる。  
**CTRL** **D** のとき“つ”は“っ”になる。

- ⑭ **CTRL** - **J**、**CTRL** - **Q**、**J** のとき1行インサート、**Q** のとき1行デリート、動作できないときはブザーが鳴る。  
 ⑮ **CTRL** - **T**、カーソルから右の文字列を次の行へ連結する。連結して1行に納まらないときはブザーが鳴ってカーソルが次の行の中途に現われるので、ここでもう一度 **CTRL** - **T** を押すと連結できる長さになる。  
 ⑯ **CTRL** - **V**、カーソルのある行をセンタリング。  
 ⑰ **CTRL** - **W**、カーソルのある行の文字の種類（形状）指定（0～4の数を入力）。  
 ⑱ **CTRL** - **P**、プリント、タブ停止位置セット、コード入力、漢字入力方法の指定のとき使用、プリントでは文字と文字の間の空白、紙送量（いずれもドット単位）、縦書きか横書きかを入力します。省略（**RETURN** キーだけ押す）した場合は **GAP** = 0、**FEED** = 12、横書きと指定したことになります。  
 ⑲ **CTRL** - **C**、プログラム終了。

## ④漢字の入力

- ① **ESC** で漢字モードにする。  
 ② 音読みでカナ2文字を入力する（1文字めを入力したと

き左下の四角の中に入力したカナが表示される）。

- ③ 同じ読みの漢字が複数あるときは、画面下部に表示された中から **↑** **↓** **←** **→** で選択して **RETURN** キーで入力する（なかった場合はスペース・キーを押す）。

ただし、**CTRL** - **P** で漢字の入力方法をTYPE2に指定していると④の部分のようになる。

- ④ **SPACE** キー、**BS** で選択し **RETURN** キーで入力します。（なければ **↓** を押す）。

もし、カナ1文字めを誤って入力した場合などは **ESC** を一度押して漢字モードを一度OFFにして入力し直してください。

“んん”、“をき”、“んき”、“をろ”と“んろ”で、それぞれ記号入力、ギリシア文字、ロシア文字入力ができます。

## ⑤ローマ字の入力

- ① **CTRL** - **G** でローマ字モードをONにする。  
 ② 日本語綴りで入力します（1文字めを入力したとき左下の四角の中に入力した文字が表示）。  
 このルーチンでは日本式かつカナ1文字に対応するものしか入力できない。ただし、“だ”、“は”などは1文字として扱うので入力できます。また、母音または“N”（ん）または **CTRL** - **G** が入力されるまで入力待ちを続けるので、母音とその1文字が正しければ、何度キーを押してもよいし“A”=“あ”、“I”=“い”と対応するので1文字めの入力を誤った場合でも簡単に訂正できます。

漢字入力と組み合わせれば、アルファベット4文字→漢字変換になります。

# IV 印字する文字の大きさ方向の指定

III 3 ⑯の **CTRL** - **W** で入力した数は印字するときの文字の大きさや方向の指定になる。

- 0…標準。  
 1…縦横反転。これを指定した行は **CTRL** - **P** の0で横書きを指定すれば縦に、縦に指定すれば横に印字する。  
 2…紙送り方向拡大。  
 3…1と2を同時に指定した状態。  
 4…倍角度指定、その行の文字を倍角にする。

# V 特殊記号

- ① 倍角記号…“↗”  
 この記号に続く1文字を倍角。  
 ② 改行記号…“↓”  
 この記号がある行ではこの記号の数だけ余計に改行する。  
 ③ 改行禁止記号…“■”  
 この記号のある行では、改行。  
 ④ 改項記号…“↔”  
 この記号があると改項この記号のある行は印字しないので注意。  
 \* ①の記号が2つ以上続くと“ERR DATA”になります。  
 \* ②、③の記号は文字列の後端にしか置けず、これ以外の場合の動作は保障していません。  
 \* これらの記号は、普通の文字同様にエディットできます。入力には **GRAPH** - **7**、**5**、**8**、**9** を使用。  
 \* これらの記号を印字するときは、そのコードに\$80を加えたものをコード入力を使って入力。  
 応用として“↑”を利用すると、疑似アンダーライン、

## 出力例(原寸)

## 1. 概要.

このプログラムは、次のような項目を実現しようと作成しました。

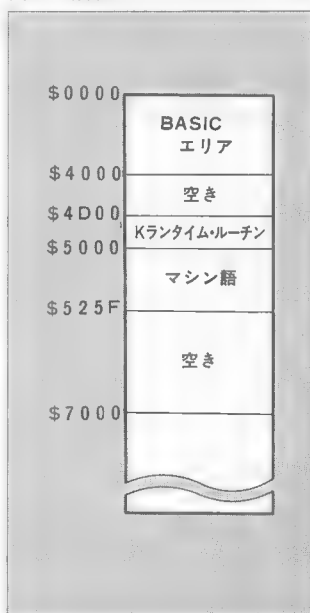
- 誰にでも使用できるように、操作を簡単にする。
- カセットベースでも使用できるようにする。
- 待ち時間が気にならないくらいまで高速化する。
- プリントは漢字プリンタを使用しないで、実用に耐え得るものとする。
- ローマ字入力を可能とする。

aを満たすため上下・左右スクロールを可能にしたり、cを満足するためにプログラム本体をKで作成した他、サブシステムに漢字表示や、スクロールプログラムを転送したりしています。dについてはMP-80TYPE2用になっていますので、LP〔バッファアドレス、文字間空白ドット数×256+出力文字数、字の種類×256〕の引数でプリントするサブルーチンを、\$4150からの0.5Kへプリンタに合わせて作成すれば、他のプリンタでも使用できます。

図4 メモリ・マップ



図6 消書プログラムメモリ・マップ



います。参考までにソース・リスト(一部)をリスト5に示しておきますが、このソース・プログラムはFM-8のカセットベースKコンパイラ・ファイル版以外ではコンパイルできないので注意してください。また、\$4D00番地から\$4FFF番地までのKコンパイ

ラのランタイム・ルーチンは発表されているものに一部変更を加えてあるので、リストに掲載されているものを使うようにしてください。それから図4と図5にメモリ・マップとデータ・ファイルのフォーマットを示しておきます。

異なる字種の混合などができる。

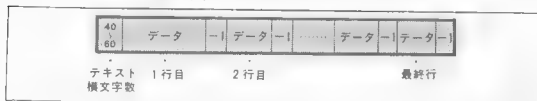
## 「VI プログラムについて」

このプログラムの主要部分はKコンパイラで記述されて

# Ⅶ 清書プログラム

このプログラムは本ワード・プロセッサで作成した複数のテキスト・ファイルを連続して印字するときに使うものです。

図5. テープのフォーマット



リスト1 日本語ワープロBASICプログラム

```
1000 CLEAR 20,&H144F:DEFINT A-Z:DIM O(60):WIDTH 80,25:CONSOLE 24,1:BADR=&H50A7
1010 LINE INPUT"HOT OR COLD (H/C)":IN$:IF IN$="h" OR IN$="H" THEN 1030
1020 PRINT "program loading":LOADM"JWP M":IF PEEK(&HFBFA)=&HF2 THEN 1028
1025 POKE &H5170,&H1F:POKE &H517E,&H1F
1028 PRINT:EXEC BADR
1030 POKE&H4BAF,BADR*256+POKE&H4BB0,BADR MOD 256
1040 FOR I=1 TO 10:KEY I,CHR$(&H15)+CHR$(I):NEXT
1050 LINE INPUT"DATA LOAD (Y/N)":IN$:IF IN$="Y" OR IN$="Y" THEN 1080
1060 INPUT"コ(40..60) ":EM:IF EM<40 OR EM>60 THEN 1060 ELSE GOSUB 1220
1070 K=TEXT-2:I=BADR+2:L=0:GOSUB 1210:GOTO 1090
1080 LINE INPUT"FILE NAME = ":IN$:PRINT"data loading":GOSUB 1160
1090 POKE BADR+1,EM:EXEC &H1450:CONSOLE 0,25:LINE INPUT"SAVE (Y/N) ":IN$
1100 IF IN$="Y" OR IN$="Y" THEN LINE INPUT"FILE NAME = ":IN$:ELSE END
1110 OPEN"O",#1,IN$:PRINT#1,EM:K=PEEK(BADR+2)*256+PEEK(BADR+3):I=TEXT:J=0
1120 WHILE I<=K:E=0
1130 FOR L=0 TO EM-1:O(L)=PEEK(I+L)*256+PEEK(I+L+1):IF O(L)<>0 THEN E=L
1140 NEXT:PRINT#1,PEEK(BU1+J):J=J+1
1150 FOR L=0 TO E:PRINT#1,O(L):NEXT:PRINT#1,-1:I=I+EM2:WEND:END
1160 OPEN"I",#1,IN$:INPUT#1,EM:GOSUB 1220:I=TEXT:J=0
1170 WHILE NOT(EOF(1)):L=0:INPUT#1,K:POKE BU1+J,K:J=J+1
1180 INPUT#1,K:IF K=-1 THEN K=0 ELSE GOSUB 1210:L=L+1:GOTO 1180
1190 FOR L=L TO EM-1:GOSUB 1210:NEXT:I=I+EM2
1200 WEND:K=I-1:I=BADR+2:L=0:GOSUB 1210:CLOSE:RETURN
1210 POKE(I+L),K*256+POKE(I+L+1),K MOD 256:RETURN
1220 BU1=BADR+916:ASZ=&H6FFF-BU1:EM2=EM*2:SN=ASZ*EM2-10
1230 IF (ASZ MOD EM2)<SN+10 THEN SN=SN-1
1240 POKE BADR+4,SN:TEXT=BU1+SN+10:FOR K=BU1 TO BU1+SN+9:POKE K,0:NEXT:RETURN
```

リスト2 日本語ワープロBASICプログラム

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1450	34	7F	C6	4C	1F	9B	CE	4C	00	CC	00	07	DD	00	CC	00	:15
1460	1B	DD	02	CC	00	26	DD	04	CC	00	27	DD	06	CC	22	2C	:BD
1470	DD	08	CC	22	2A	DD	0A	CC	22	2B	DD	0C	4F	F6	4B	AF	:25
1480	36	06	CC	01	00	BD	4D	1F	36	06	4F	F6	4B	80	E3	C1	:52
1490	DD	0E	CC	0E	C3	03	1C	DD	0C	0E	C3	03	94	DD	12	:07	
14A0	DD	0E	C3	00	04	1F	01	4F	E6	84	DD	14	DC	0E	D3	14	:4C
14B0	C3	03	9E	DD	16	DC	0E	C3	00	06	DD	1B	DC	0E	C3	00	:AC
14C0	10	DD	1A	DC	0E	C3	00	01	1F	01	4F	E6	84	DD	1C	DC	:63
14D0	1C	93	06	83	00	01	DD	1E	DC	06	93	1E	83	00	01	DD	:28
14E0	20	CC	00	00	DD	22	CC	00	00	DD	24	CC	00	00	DD	26	:87
14F0	CC	00	00	DD	28	CC	00	08	DD	2A	CC	00	00	DD	2C	CC	:4D
Sum:	F6	C5	BD	62	39	0B	D6	51	F2	71	ED	A5	3F	DC	B5	6D	:77

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1500	FF	FF	DD	2E	CC	00	00	DD	30	CC	00	00	DD	32	CC	00	:69
1510	00	DD	34	CC	00	00	DD	36	CC	00	00	DD	38	CC	00	00	:9D
1520	DD	3A	CC	06	83	00	01	DD	3C	CC	0E	C3	00	01	DD	3E	:6F
1530	CC	00	07	DD	40	CC	00	00	DD	42	CC	00	09	36	06	CC	:C8
1540	42	16	00	14	DC	42	9E	18	30	8B	36	10	CC	00	00	E7	:F4
1550	D1	DC	42	C3	00	01	DD	42	A3	C4	2F	E8	33	42	DC	0E	:AF
1560	C3	00	02	1F	01	4F	E6	84	36	06	CC	01	00	BD	4D	1F	:D0
1570	36	06	CC	0E	C3	00	03	1F	01	4F	E6	84	E3	C1	93	16	:12
1580	DD	44	DC	44	47	56	C3	00	01	DD	42	DC	3C	36	06	CC	:E1
1590	00	09	36	06	CC	00	01	36	06	CC	14	34	06	37	10	34	:F3
15A0	10	37	10	34	10	37	10	34	10	17	E1	32	68	36	06	11	:13
15B0	DC	42	16	00	16	DC	42	58	49	9E	16	30	8B	36	10	CC	:8A
15C0	00	00	ED	D1	DC	42	C3	00	01	DD	42	A3	C4	2F	E6	33	:9E
15D0	42	17	1D	1B	17	26	EA	BD	40	DD	46	17	26	E2	DC	:86	
15E0	46	83	00	7F	10	26	00	03	17	22	DF	DC	46	83	00	08	:46
15F0	10	26	00	0D	DC	22	83	00	01	DD	22	17	21	C0	17	22	:F5
Sum:	15	94	56	D7	47	77	8B	6F	E5	AE	AA	1A	41	9B	A6	4F	:80

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1600	C9	DD	46	83	00	1C	10	26	00	0A	2B	C3	00	01	DD	:69	
1610	22	17	21	AA	DC	46	83	00	1D	10	26	00	0A	DC	22	83	:B7
1620	00	01	DD	22	17	21	97	DC	46	83	00	1E	10	26	00	0A	:D2
1630	DC	24	83	00	01	DD	24	17	21	84	DC	46	83	00	1F	10	:15
1640	26	00	0A	DC	24	C3	00	01	DD	24	17	21	71	DC	46	83	:43
1650	00	02	10	26	00	0A	DC	22	83	00	05	DD	22	17	21	5E	:80
1660	DC	46	83	00	06	10	26	00	0A	DC	22	C3	00	05	DD	22	:BD
1670	17	21	48	DC	46	83	00	19	10	26	00	22	DC	22	83	00	:1A
1680	00	10	26	00	0A	DC	24	83	00	03	DD	24	16	00	CC	:B5	
1690	00	0D	DC	22	DC	24	83	00	01	DD	24	17	21	20	DC	46	:FE
16A0	83	00	1A	10	26	00	22	DC	22	83	00	00	10	26	00	0A	:B6

16B0	DC	24	C1	00	93	DD	24	16	00	0C	CC	00	00	DD	22	DC	:90
16C0	24	C3	00	01	DD	24	17	20	F5	DC	46	83	00	16	10	26	:06
16D0	01	36	CC	FF	FF	DD	42	CC	00	00	DD	4B	CC	00	00	36	:13
16E0	06	01	24	36	06	CC	00	00	36	06	CC	2B	34	06	37	10	:CF
16F0	34	10	37	10	34	10	37	10	34	10	17	1C	90	32	6B	DD	:94
Sum:	9E	9A	B6	A5	89	7A	CD	C6	80	A8	FF	B3	A6	BD	C2	BE	:B6

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1700	4A	DD	3E	36	06	CC	24	36	06	CC	00	01	36	06	CC	28	:E7
1710	3A	06	37	10	34	10	37	10	34	10	37	10	34	10	17	1C	:0E
1720	6C	32	6B	DD	4C	DC	4A	DD	4C	DC	4C	36	06	CC	44	16	:10
1730	00	4F	DC	44	9E	16	58	49	EC	8B	DD	4E	DC	4E	36	06	:CC
1740	CC	00	00	BD	4D	C1	36	06	CC	4E	36	06	CC	22	2D	BD	:11
1750	4D	C1	A4	C0	E4	C0	36	06	CC	4E	36	06	CC	0B	BD	4D	:A6
1760	C1	A4	C0	E4	C0	10	27	00	11	DC	44	DD	4B	DC	42	83	:F7
1770	FF	FF	FF	10	26	00	04	DC	44	DD	42	DC	44	C3	00	01	:D8
1780	44	A3	C4	2F	AD	33	42	DC	42	83	FF	FF	10	27	00	78	:4A
1790	DC	4A	D3	4C	47	56	DD	4A	DC	42	D3	4B	47	56	DD	50	:0C
17A0	CC	FF	FF	DD	3B	CC	00	00	DD	22	17	15	24	DC	50	93	:B9
17B0	4A	10	2F	00	23	DC	4A	DD	4C	DC	50	83	00	01	36	06	:E7
17C0	DC	4C	16	00	0A	17	21	02	DC	4C	C3	00	01	DD	4C	A3	:3A
17D0	C4	2F	F2	33	42	16	00	2C	DC	50	DD	4C	DC	4A	83	00	:9A
17E0	01	36	06	CC	4C	16	00	16	CC	00	00	34	06	17	1E	14	:E0
17F0	02	62	CC	00	00	DD	22	DC	4C	C3	00	01	DD	4C	A3	C4	:DB
Sum:	CC	D6	CA	55	FC	C4	18	DF	27	1F	C5	22	3A	2A	BD	A6	:3C

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1800	2F	E6	33	42	DC	3A	DD	3B	DC	46	83	00	14	10	26	02	:A6
1810	EA	CC	00	00	DD	2E	DC	24	D3	2B	36	06	CC	00	09	D3	:A0
1820	14	BD	4D	A1	10	27	02	CE	CC	FF	FF	DD	3B	DC	22	DD	:80
1830	4C	DC	26	83	00	00	10	27	00	06	CC	4C	D3	3E	DD	4C	:70
1840	CC	FF	FF	DD	4A	CC	FF	FF	DD	50	CC	00	00	36	06	CC	:CC
1850	24	C2	00	01	36	06	CC	00	00	36	06	CC	2B	34	06	37	:A1
1860	10	34	10	37	10	34	10	37	10	34	10	17	1B	1F	1C	32	:68
1870	DD	4B	DC	4B	DD	44	DC	22	36	06	CC	24	36	06	CC	2B	:E2
1880	36	06	CC	2B	34	06	37	10	34	10	37	10	34	10	37	10	:D7
1890	34	10	17	1A	FB	32	6B	DD	42	DC	3C	36	06	CC	24	36	:B0
18A0	06	CC	00	01	36	06	CC	2B	34	06	37	10	34	10	37	10	:1F
18B0	34	10	37	10	34	10	17	1A	D4	32	68	36	06	CC	42	16	:D6
18C0	00	8B	DC	42	9E	16	58	49	EC	8B	DD	4E	DC	4E	36	06	:06
18D0	CC	00	00	BD	4D	C3	36	06	CC	4E	36	06	CC	22	2D	BD	:09
18E0	4D	B9	AA	C0	EA	C0	36	06	CC	4E	36	06	CC	0B	BD	4D	:AA
18F0	B9	AA	C0	EA	C0	10	27	00	3C	DC	4E	36	06	CC	22	2D	:C1
Sum:	CC	69	01	BF	61	C6	FF	2D	FC	5A	FB	62	62	D5	5E	4B	:DB

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1900	BD	4D	B9	36	06	DC	4E	36	06	DC	08	BD	4D	B9	AA	C0	:76
1910	EA	C0	10	27	0F	DC	42	58	49	9E	16	30	8B	36	10	:64	
1920	CC	00	00	ED	D1	CC	4A	83	FF	FF	10	26	00	0A	DC	:B9	
1930	DD	04	DC	42	48	DC	42	C3	00	01	DD	42	A3	C4	10	:BF	
1940	00	04	DC	42	48	DC	42	C3	00	01	DD	42	A3	C4	10	:BF	
1950	2F	4F	4F	43	42	DC	4A	83	FF	FF	10	27	00	12	DC	:A8	
1960	93	48	4D	48	4C	93	4A	83	00	01	DD	4A	16	00	06	:C4	
1970	DC	44	93	48	4D	48	CC	00	00	DD	50	DC	4A	36	06	:57	
1980	50	16	00	11	CC	00	00	34	06	10	17	10	78	32	62	:50	
1990	C3	00	01	DD	50	A3	C4	2F	EB	33	42	17	13	32	CC	:00	
19A0	00	DD	22	DC	24	C3	00	01	DD	24	17	1E	11	DC	C3	:36	
19B0	06	DC	24	36	06	CC	00	01	34	06	DC	28	34	06	37	:10	
19C0	34	10	37	10	34	10	37	10	34	10	17	10	00	32	68	:C1	
19D0	4A	CC	00	00	DD	50	DC	50	36	06	DC	48	BD	4D	A1	:36	
19E0	06	DC	4A	9E	16	58	49	EC	8B	36	06	CC	00	BD	4D	:0A	
19F0	B9	A4	C0	E4	C0	10	27	00	B2	DC	4A	93	1C	9E	16	:58	
Sum:	44	11	22	E1	ED	3D	3F	BA	2A	E6	8B	93	09	21	69	:C2	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1A00	49	EC	B8	34	06	17	1B	FC	32	62	DC	4A	93	1C	DD	:52	:E8
1A10	DC	4A	93	1C	93	1C	D3	CC	00	00	36	06	DC	52	16	:E8	
1A20	00	33	CC	FF	FF	DC	52	58	49	9E	16	30	8B	36	10	:00	
1A30	52	83	00	01	9E	16	58	49	EC	8B	ED	D1	CC	50	C3	:00	
1A40	01	58	49	9E	16	30	8B	36	10	34	10	17	10	00	DD	:11	
1A50	C3	FF	FF	DC	52	A3	C4	2C	9C	33	42	CC	00	00	DD	:52	
1A60	DC	3C	36	06	DC	52	16	00	32	DC	52	36	06	DC	24	:83	
1A70	00	01	36	06	CC	00	00	13	93	1C	D3	52	03	00	01	:9E	
1A80	16	58	49	EC	8B	34	06	37	10	34	10	17	10	34	10	:93	
1A90	19	23	32	66	DC	52	C3	00	01	DD	50	DC	C4	2F	CA	:76	
1AA0	42	CC	50	C3	00	01	DD	50	16	FF	2B	DC	3A	DD	38	:A8	
1AB0	50	93	48	10	2C	00	2E	CC	00	07	BD	4E	DE	DC	3C	:93	
1AC0	48	D3	50	C3	00	01	DD	22	CC	00	00	10	2D	00	:EC		
1AD0	08	CC	FF	FF	DC	26	17	16	7F	16	00	06	CC	22	D3	:3E	
1AE0	DD	22	16	00	11	CC	4A	9E	16	58	49	9E	16	30	8B	:36	
1AF0	10	CC	00	00	ED	D1	CC	FF	FF	DD	2E	DC	46	83	00	:17	
Sum:	18	F7	16	BE	C4	EF	6E	84	65	06	8C	59	DA	2E	79	:4D	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1B00	10	E2	DD	0A	DC	4A	83	00	07	10	2F	91	41	CC	00	:00	
1B10	4D	E2	DD	0A	DC	4A	83	00	07	10	2F	91	41	CC	00	:00	
1B20	DD	50	CC	00	00	DD	22	17	11	A7	CC	00	00	36	06	:0A	
1B30	24	36	06	CC	00	00	36	06	DC	28	34	06	37	10	34	:10	
1B40	37	10	34	10	37	10	34	10	17	18	42	32	68	DD	48	:C1	
1B50	FF	FF	DC	38	CC	00	00	DD	4C	DC	22	36	06	DC	3C	:DD	
1B60	4D	A1	36	06	DC	26	36	06	CC	00	BD	40	B9	AA	C0	:61	
1B70	EA	C0	E4	C0	10	27	00	CC	00	DC	18	83	00	01	BD	:4A	
1B80	4A	C0	E4	C0	10	27	00	CC	00	DC	18	83	00	01	BD	:4A	
1B90	EC	8B	DD	54	DC	54	36	06	CC	00	BD	40	B9	36	06	:0A	
1BA0	DC	54	36	06	CC	22	20	BD	40	B9	AA	C0	E4	C0	:36		
1BB0	DC	54	36	06	CC	00	BD	40	B9	AA	C0	E4	C0	:36			
1BC0	54	36	06	DC	0A	BD	40	B9	AA	C0	E4	C0	:36				
1BD0	D3	1C	83	00	01	9E	16	58	49	EC	8B	36	06	CC	:00		
1BE0	BD	4D	A1	AA	C0	E4	C0	10	27	00	54	DC	48	:65			
1BF0	D3	4C	83	00	01	9E	16	58	49	EC	8B	36	06	DC	:0A		
Sum:	CA	DC	27	AA	B4	54	DF	6B	8B	4F	A8	52	48	28	41	:11	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1C00	4D	C1	36	06	DC	4C	36	06	CC	00	BD	40	B9	AA	C0	:A7	
1C10	EA	C0	10	27	00	CC	00	22	93	3C	10	2F	00	10	CC	:00	
1C20	01	DD	26	17	16	C9	DC	22	93	06	03	20	DD	22	CC	:0A	
1C30	34	06	17	19	CF	32	62	DC	4C	C3	00	01	DD	4C	22	:E0	
1C40	DD	50	CC	00	22	C3	00	01	DD	22	CC	4C	C3	00	01	:00	
1C50	16	FF	06	DC	50	DD	22	CC	3A	DD	38	16	00	40	DC	:24	
1C60	D3	28	9E	12	E6	8B	4F	83	00	00	10	27	00	09	DC	:24	
1C70	34	06	17	1E	02	32	62	DC	24	83	28	1E	13	30	8B	:3A	
1C80	10	DC	4A	E7	01	DC	24	38	9E	12	E6	8B	4F	83	00	:DC	
1C90	00	10	27	00	09	DC	24	34	06	17	1D	DB	32	62	DC	:46	
1CA0	83	00	18	10	26	01	CB	BD	40	0B	46	55	4E	43	54	:49	
1CB0	4F	E4	20	48	45	59	20	4E	55	4D	42	45	52	20	3D	:0C	
1CC0	00	BD	4D	E2	DD	46	DC	46	36	06	CC	00	BD	40	A1	:EF	
1CD0	36	06	DC	46	36	06	CC	00	00	BD	40	A9	A4	C0	E4	:C0	
1CE0	10	27	00	83	DC	46	83	00	01	DD	48	CC	00	DD	50	:E4	
1CF0	17	0F	CC	DC	46	83	00	11	10	26	00	52	DC	04	36	:06	
Sum:	A5	14	B4	B4	36	34	B2	A7	D5	64	B7	CD	01	46	B2	:1C	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1D00	CC	48	BD	4D	1F	D3	56	48	49	9E	1A	30	8B	36	10	:DC	
1D10	22	83	00	01	36	06	DC	24	36	06	DC	26	36	06	DC	:28	
1D20	34	06	37	10	34	10	37	10	34	10	37	10	34	10	17	:08	
1D30	5C	32	68	9E	16	58	49	EC	8B	ED	D1	CC	50	C3	00	:01	
1D40	DD	50	CC	00	48	9E	18	30	8B	36	10	DC	50	E7	D1	:CC	
1D50	36	06	CC	00	01	BD	40	C1	36	06	DC	50	36	06	DC	:04	
1D60	BD	4D	99	AA	C0	E4	C0	10	27	00	87	DC	46	83	00	:11	
1D70	00	1B	CC	00	07	BD	4E	DE	17	1F	46	BD	4D	D6	DD	:46	
1D80	17	1F	3E	CC	46	83	00	11	27	00	16	00	07	DC	22	:83	
1D90	00	01	DD	22	16	00	07	1F	A2	CC	00	01	DD	42	CC	:7C	
1DA0	00	0A	36	06	DC	42	16	00	4C	CC	00	00	36	06	DC	:42	
1DB0	06	CC	00	0A	BD	4D	38	36	06	CC	00	0A	BD	4D	1F	:A3	
1DC0	C1	53	43	C3	00	01	DD	52	CC	42	83	00	01	DD	54	:CC	
1DD0	00	00	36	06	DC	42	36	06	CC	00	12	BD	4D	1F	83	:00	
1DE0	12	36	06	DC	52	C3	00	30	36	06	CC	00	04	34	06	:37	
1DF0	10	34	10	37	10	34	10	37	10	34	10	17	1F	62	32	:68	
Sum:	5E	74	49	D8	48	09	B1	E6	2E	60	D7	3E	EB	41	7C	:64	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
1E00	CC	00	01	DD	4A	DC	54	9E	1B	E6	8B	4F	36	06	DC	:44	
1E10	16	00	35	DC	4A	36	06	DC	42	83	00	01	36	06	DC	:44	
1E20	36	06	DC	54	9E	16	00	4D	1F	D3	4A	83	00	01	9E	:1A	
Sum:	5E	74	49	D8	48	09	B1	E6	2E	60	D7	3E	EB	41	7C	:64	

1E30	EC	8B	34	06	37	10	34	10	37	10	34
------	----	----	----	----	----	----	----	----	----	----	----

## リスト 2

2370	36	06	DC	44	16	00	56	DC	54	DC	44	9E	16	58	49	EC	:50
2380	88	DD	4E	DC	4E	83	00	00	10	27	00	3A	DC	4E	83	22	:A3
2390	2D	10	27	00	2A	DC	4E	93	08	10	26	00	08	CC	00	00	:5D
23A0	DD	52	16	00	17	DC	4E	93	0C	10	26	00	08	CC	FF	FF	:2D
23B0	DD	5E	16	00	07	DC	44	C3	00	01	DD	4A	16	00	07	DC	:5C
23C0	52	C3	00	01	DD	52	DC	44	C3	00	01	DD	4A	A3	C4	2F	:E0
23D0	A6	33	42	DC	42	9E	12	E6	8B	4F	DD	60	DC	5A	83	00	:9F
23E0	01	10	26	00	29	DC	60	26	06	DC	60	47	56	58	49	A3	:F5
23F0	C1	53	43	C3	00	01	83	00	00	10	26	00	0A	DC	60	C3	:DD
Sum:	2F	CC	53	CC	05	07	DB	D4	83	62	D1	46	D9	78	E9	51	:58
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2400	00	01	DD	60	16	00	07	DC	60	83	00	01	DD	60	DC	54	:88
2410	D3	4A	83	00	01	9E	16	58	49	EC	8B	36	06	DC	0A	DD	:4C
2420	4D	B9	36	06	DC	4A	76	06	CC	00	00	BD	4D	A9	A4	C0	:87
2430	E4	C0	10	27	00	07	DC	4A	83	00	01	DD	4A	DC	5E	10	:FD
2440	27	00	00	CC	00	0C	34	06	BD	4B	93	62	16	00	87	:12	
2450	CC	00	00	DD	50	DC	4A	83	00	01	DD	4A	C3	00	16	00	:21
2460	1F	DC	5C	D3	00	58	49	9E	10	30	8B	36	10	DC	54	D3	:CD
2470	50	9E	16	58	49	EC	8B	ED	D1	DC	50	C3	00	01	DD	50	:F7
2480	A3	C4	2F	DC	33	42	DC	10	36	06	DC	56	36	06	CC	01	:A8
2490	00	BD	4D	1F	D3	4A	D3	5C	36	06	DC	50	36	06	CC	01	:F6
24A0	00	BD	4D	1F	34	06	17	10	34	10	34	10	34	10	34	10	:46
24B0	B4	32	66	CC	00	01	DD	44	DC	52	36	08	DC	44	16	00	:AA
24C0	19	DC	58	34	06	17	07	BF	32	62	DC	44	C3	00	01	DD	:B0
24D0	44	A3	C4	2F	DC	33	42	DC	10	C3	00	01	DD	42	A3	C4	:A3
24E0	10	2F	FE	4D	CC	33	42	CC	00	00	DD	46	DC	22	36	06	:D4
24F0	24	36	06	DC	26	36	06	DC	2B	34	06	37	10	34	10	37	:9E
Sum:	15	92	74	D4	61	70	5E	CF	AE	6B	7D	26	16	10	54	87	:AB
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2500	10	34	10	37	10	34	10	17	0E	83	22	68	DD	42	DC	46	:62
2510	36	06	CC	00	20	BD	4D	A9	36	06	DC	46	36	06	CC	00	:41
2520	41	BD	4D	A1	A4	C0	E4	C0	36	06	DC	46	36	06	CC	00	:5A
2530	5A	BD	4D	A9	36	06	DC	46	36	06	CC	00	61	BD	4D	A1	:7F
2540	A4	C0	E4	C0	E4	C0	E4	C0	10	27	00	1B	DC	46	36	06	:C9
2550	CC	01	00	BD	4D	1F	34	06	00	4B	93	62	16	00	87	:19	
2560	10	A2	32	62	DC	46	36	06	CC	00	70	BD	4D	A9	36	06	:D9
2570	DC	46	36	06	CC	00	70	BD	4D	A1	A4	C0	E4	C0	36	06	:98
2580	DC	32	36	06	CC	00	00	BD	4D	A9	36	06	DC	46	36	06	:73
2590	CC	00	20	BD	4D	B9	A4	C0	E4	C0	36	06	CC	00	10	27	:02
25A0	00	18	DC	46	36	06	CC	01	00	BD	4D	1F	34	06	BD	48	:AB
25B0	83	32	62	34	06	17	10	4C	32	62	DC	46	36	06	CC	00	:82
25C0	7F	BD	4D	A9	36	06	DC	46	36	06	CC	00	AA	BD	4D	A1	:E9
25D0	AA	C0	E4	C0	36	06	DC	46	36	06	CC	00	BD	4D	BD	B9	:E1
25E0	AA	C0	E4	C0	36	06	DC	46	36	06	CC	00	DD	4D	BD	A9	:0A
25F0	36	06	DC	46	36	06	CC	00	FF	BD	4D	A1	A4	C0	E4	C0	:19
Sum:	6B	1C	4D	12	D6	CA	D0	EB	9A	0C	11	B7	20	F7	03	48	:E1
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2600	AA	C0	E4	C0	10	27	00	18	DC	46	36	06	CC	01	00	BD	:48
2610	4D	1F	34	06	BD	48	83	32	62	34	06	17	0F	E6	32	62	:9C
2620	DC	46	36	06	CC	00	AA	BD	4D	A9	36	06	DC	46	36	06	:1D
2630	CC	00	BD	4D	A1	A4	C0	E4	C0	10	27	00	1A	DC	46	:A2	
2640	36	06	CC	01	00	BD	4D	1F	34	06	34	06	BD	48	83	32	:EF
2650	62	34	06	17	0F	AE	32	62	DC	46	83	00	09	10	26	00	:E8
2660	4D	CC	00	00	BD	3B	DC	22	36	06	DC	2A	BD	4D	3B	36	:E9
2670	06	DC	2A	BD	4D	1F	D3	2A	BD	4D	4A	DC	A9	37	3C	10	:2F
2680	00	04	DC	3C	DD	4A	DC	22	DD	50	DC	4A	36	06	DC	50	:FC
2690	16	00	11	CC	00	00	34	06	17	0F	69	32	62	DC	50	C3	:3F
26A0	00	01	DD	50	A3	C4	2F	DC	33	42	DC	3A	DD	3B	DC	46	:71
26B0	83	00	0A	10	26	01	BD	DC	3C	36	06	CC	00	09	36	06	:A9
26C0	CC	00	01	36	06	DC	14	34	06	17	10	34	10	37	10	34	:39
26D0	10	37	10	34	10	17	0C	85	32	68	DD	44	CC	00	00	DD	:D7
26E0	42	DC	44	93	1C	C3	00	01	DD	52	DC	44	36	06	DC	52	:8E
26F0	16	00	1D	DC	52	9E	16	58	49	EC	8B	83	00	00	10	27	:E7
Sum:	57	1F	46	9F	49	35	F0	C5	B2	63	6C	85	54	8B	72	EB	:CD
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2700	00	07	DC	44	C3	00	01	DD	42	DC	52	C3	00	01	DD	52	:28
2710	A3	C4	2F	DC	33	42	DC	10	36	06	DC	10	2F	00	09	:16	
2720	CC	00	07	BD	4E	DE	16	01	0E	CC	00	00	36	06	DC	24	:E9
2730	C3	00	01	36	06	CC	00	00	10	26	06	DC	2B	34	06	37	:8D
2740	34	10	37	10	34	10	37	10	17	0C	A3	32	68	DD	:34		
2750	5C	DC	5C	DC	5A	DC	44	DC	52	DC	5C	DC	16	30	8B	36	:C6
2760	52	15	00	20	CC	FF	FF	DC	52	DC	58	49	9E	16	30	8B	:3A
2770	10	DC	52	93	1C	9E	16	58	49	EC	8B	ED	D1	DC	52	C3	:68
2780	FF	FF	DC	52	A3	C4	2F	DC	33	42	DC	5C	DC	50	DC	5C	:80
2790	D3	1C	83	00	01	36	06	DC	52	16	00	16	DC	52	58	49	:D8
27A0	9E	16	30	8B	36	10	CC	00	ED	D1	DC	52	C3	00	01	:31	
27B0	DD	52	A3	C4	2F	DC	E6	33	42	DC	22	36	06	DC	24	36	:96
27C0	DC	26	36	06	DC	2B	34	06	37	10	34	10	37	10	34	10	:92
27D0	37	10	34	10	17	0B	86	32	68	DD	42	DC	42	DC	52	DC	:45
27E0	5E	93	00	01	36	06	DC	52	16	00	31	DC	54	58	49	9E	:00
27F0	16	30	8B	36	10	DC	52	9E	16	58	49	EC	8B	ED	D1	DC	:AB
Sum:	F6	15	20	A4	FC	7A	CC	63	A6	A6	DB	A1	FC	6D	45	53	:3D
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2800	52	58	49	9E	16	30	8B	36	10	CC	00	ED	D1	DC	54	:62	
2810	C3	00	01	DD	54	DC	52	C3	00	01	DD	52	A3	C4	2F	DC	:77
2820	33	42	17	0F	1D	DC	24	C3	00	01	DD	24	CC	00	00	DD	:26
2830	22	17	0F	8A	17	04	9A	DC	46	83	00	0F	10	26	01	21	:93
2840	DC	2B	36	06	CC	00	00	BD	4D	A9	36	06	DC	24	36	06	:07
2850	CC	00	00	BD	4D	A9	AA	CC	EA	C0	10	27	00	FD	CC	00	:93
2860	00	36	06	DC	24	36	06	CC	00	00	36	06	DC	2B	34	06	:BE
2870	37	10	34	10	37	10	34	10	37	10	34	10	17	0B	0E	32	:03
2880	68	DD	62	DC	42	93	62	10	27	00	3A	CC	00	01	DD	4C	:21
2890	DC	22	36	06	DC	26	36	06	CC	06	80	4D	1F	A3	E1	53	:3

28B0	DD	22	17	0F	09	17	10	12	DC	4C	C3	00	01	DD	4C	A3	:1F	
28C0	C4	2F	E8	33	42	DC	3C	36	06	CC	00	09	36	06	CC	00	:81	
28D0	01	36	06	DC	14	34	06	37	10	34	10	37	10	34	10	:37	:84	
28E0	10	34	10	17	0A	F7	32	68	DD	44	DC	62	DD	52	DC	44	:64	
28F0	36	06	DC	52	16	00	1D	DC	52	93	1C	58	A9	9E	16	30	:FF	
Sum:	BB	A2	69	2D	E5	68	94	16	FE	F3	40	B7	E9	3D	08	49	:46	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
2900	BB	36	10	00	DD	52	9E	16	58	49	EC	BB	ED	D1	DC	52	C3	:7A
2910	00	00	DD	52	DC	A3	2F	DF	33	42	DC	44	93	1C	C3	00	:0C	
2920	01	DD	52	DC	44	36	06	DC	52	16	00	16	DC	52	58	49	:85	
2930	9E	16	30	BB	36	10	CC	00	00	00	D1	DC	52	C3	00	00	:17	
2940	DD	52	A3	C4	2F	E6	33	42	DC	24	83	00	01	DD	24	83	:31	
2950	0E	6C	17	0D	ED	15	73	7F	14	00	06	CC	00	07	BD	4E	:18	
2960	DE	CC	46	83	00	05	10	26	00	C4	06	CC	36	06	DC	24	:06	
2970	36	06	CC	00	01	36	06	DC	28	34	06	37	10	34	10	:47	:45	
2980	10	34	10	37	10	34	10	17	A4	03	32	68	DD	44	DC	42	:DC	
2990	DD	54	DC	44	36	06	DC	54	16	00	16	DC	54	58	49	9E	:58	
29A0	16	58	49	36	10	CC	00	00	ED	D1	DC	54	C3	00	01	DD	:72	
29B0	54	A3	C4	2F	E6	33	42	DC	46	83	00	08	10	26	00	3A	:62	
29C0	BD	4D	0B	54	DC	20	45	31	20	3C	20	3A	20	20	54	59	:23	
29D0	50	45	32	20	3C	20	32	20	00	BD	4D	E2	DD	46	DC	46	:C7	
29E0	83	00	00	00	26	00	05	CC	CC	00	DD	30	DC	46	83	00	:5D	
29F0	02	10	26	00	05	CC	FF	CC	DD	30	DC	22	DD	54	DC	3C	:5D	
Sum:	12	C7	DA	4D	89	55	0C	33	38	CE	ED	67	93	ED	EF	9F	:85	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
2A00	36	06	DC	54	16	00	21	DC	54	36	06	DC	24	36	06	CC	:17	
2A10	00	00	34	06	37	10	34	10	37	10	34	10	17	09	96	32	:93	
2A20	66	DC	54	C3	00	01	DD	54	C3	C4	2F	DB	33	42	DC	46	:93	
2A30	34	06	17	06	CC	32	62	DC	34	10	27	00	00	DC	64	36	:88	
2A40	06	DC	46	AA	CC	EA	00	10	27	00	03	17	0A	34	16	00	:01	
2A50	39	DC	64	36	06	CC	06	6A	AA	CC	EA	00	10	27	00	28	:DC	
2A60	46	36	06	CC	01	00	BD	4D	1F	36	06	CC	00	02	36	06	:BE	
2A70	DC	6B	DD	4D	1F	A3	C1	53	C3	C3	00	01	34	06	BD	48	:64	
2A80	83	32	62	34	06	17	08	7C	32	62	CC	46	36	06	CC	00	:AD	
2A90	80	BD	4D	A9	36	06	DC	46	36	06	CC	00	DE	BD	4D	A1	:52	
2AA0	AA	C0	E4	C0	36	06	DC	46	36	06	CC	00	A6	BD	4D	B9	:07	
2AB0	AA	C0	EA	C0	10	27	00	08	CC	FF	FF	DD	6A	16	00	05	:7F	
2AC0	CC	00	00	DD	6A	DC	32	10	27	00	3A	DC	6A	36	06	DC	:F0	
2AD0	46	36	06	CC	00	20	BD	4D	B9	36	06	CC	2C	A4	C0	E4	:BD	
2AE0	C0	AA	C0	EA	C0	10	27	00	06	17	02	0B	16	00	13	DC	:3A	
2AF0	46	83	00	20	10	26	00	0A	CC	00	00	34	06	17	0B	04	:55	
Sum:	CA	10	4B	2C	BB	28	11	ED	C7	B7	0E	D5	BD	20	5A	A3	:6D	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
2B00	32	62	16	00	20	DC	6A	10	27	00	1A	DC	46	36	06	CC	:8B	
2B10	01	00	DD	4D	1F	93	36	34	06	BD	48	83	32	62	34	06	:83	
2B20	17	0A	E1	32	62	CC	00	00	DD	38	DC	46	83	00	04	10	:30	
2B30	26	00	12	DC	42	9E	16	58	49	EC	BB	83	00	01	34	06	:E0	
2B40	17	0A	E1	32	62	DC	46	83	00	01	10	26	00	12	DC	42	:82	
2B50	9E	16	58	49	EC	BB	C3	00	01	34	06	17	0A	A6	32	62	:25	
2B60	DC	3A	DD	3B	DC	46	83	00	1B	10	26	00	53	17	0E	97	:30	
2B70	DC	32	34	06	17	01	69	32	62	DD	32	CC	00	3B	36	06	:9F	
2B80	CC	00	06	36	06	CC	00	43	36	06	CC	00	BE	36	06	CC	:9E	
2B90	00	01	36	06	CC	00	07	36	06	CC	00	02	34	06	37	10	:9E	
2BA0	34	10	37	10	34	10	37	10	34	10	37	10	34	10	37	10	:2C	
2BB0	34	10	37	10	34	10	17	11	E4	32	6E	CC	00	00	DD	2C	:50	
2BC0	DC	46	93	00	10	26	00	0E	17	0A	DC	34	34	06	17	:48		
2BD0	01	0E	32	62	DD	3C	46	83	00	03	10	26	00	29	BD	:78		
2BE0	4D	08	53	75	72	65	3F	2D	28	31	29	20	00	BD	4D	E2	:64	
2BF0	DD	46	DC	46	83	00	01	10	26	00	08	CC	00	03	DD	46	:F9	
Sum:	18	BE	3E	BD	40	32	1C	6F	0D	55	A6	E7	DB	E3	6E	3D	:F3	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
2C00	16	00	00	00	DD	4B	CC	00	00	DD	42	CC	3C	36	06	CC	:8F	
2C10	C3	CC	00	00	DD	4B	CC	00	00	DD	42	CC	3C	36	06	CC	:8F	
2C20	00	09	36	06	CC	00	01	36	06	CC	DD	14	34	06	37	10	:34	
2C30	10	37	10	34	10	37	10	34	10	17	07	51	32	68	36	06	:6B	
2C40	DC	42	16	00	1C	DD	42	9E	16	58	49	EC	BB	83	00	00	:BD	
2C50	10	27	00	06	DC	42	58	49	DD	48	DC	42	C3	00	01	DD	:00	
2C60	42	A3	C4	2F	E0	33	42	DC	0E	C3	00	02	36	06	DC	48	:3C	
2C70	D3	16	1F	B9	4F	E7	D1	DC	0E	C3	00	03	36	06	DC	48	:A8	
2C80	D3	16	4F	B9	4F	E7	D1	35	FF	CC	00	1B	36	06	CC	01	:00	
2C90	4D	1F	C3	00	41	34	06	BD	4B	93	32	62	EC	62	36	06	:62	
2CA0	CC	01	00	BD	4D	1F	C3	00	00	34	06	BD	4B	93	32	62	:F3	
2CB0	CC	00	0A	34	06	BD	4B	93	32	62	39	17	0B	00	17	0F	:C0	
2CC0	97	BD	4D	D6	DD	46	17	0F	BF	DC	22	C3	00	01	DD	22	:10	
2CD0	3C	DC	26	10	27	00	08	CC	00	00	DD	26	17	04	79	39	:16	
2CE0	EC	62	33	00	10	26	00	00	0B	CC	FF	FF	DD	60	16	00	:2C	
2CF0	05	CC	00	00	DD	60	39	DC	2C	83	00	00	10	26	00	0F	:17	
Sum:	63	2B	56	82	26	B2	F8	22	4E	AB	AA	B8	43	F5	16	FA	:FB	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
2D00	CC	00	01	DD	2C	DC	46	1D	CC	67	06	CC	01	00	BD	4D	:9E	
2D10	00	00	DD	2C	17	0D	16	DC	6C	36	06	CC	01	00	BD	4D	:9E	
2D20	1F	D3	46	34	06	BD	3E	00	32	62	DD	52	CC	00	00	DD	:D9	
2D30	46	DC	52	83	00	00	10	27	03	60	4F	F6	3E	20	DD	54	:65	
2D40	DC	54	DD	4C	DC	52	DD	44	DC	54	83	00	10	17	02	:92	:95	
2D50	ED	CC	30	83	00	00	10	27	00	C4	DC	46	36	06	CC	00	:A1	
2D60	00	BD	4D	C1	36	06	DC	46	36	06	CC	00	1F	BD	4D	C1	:28	
2D70	AA	C0	E4	C0	10	27	00	A3	DC	22	36	06	DC	24	36	06	:58	
2D80	DC	44	34	06	37	10	34	10	37	10	34	10	17	06	26	32	:E5	
2D90	66	DD	4D	D6	DD	46	DC	46	36	06	CC	00	1D	BD	4D	B9	:73	
2DA0	36	06	DC	46	36	06	CC	00	0B	BD	4D	B9	AA	C0	EA	C0	:45	
2DB0	10	37	00	0E	DC	44	83	00	01	DD	44	DC	54	C3	00	01	:FE	
2DC0	DD	54	DC	46	83	00	20	10	26	00	0E	DC	44	C3	00	01	:1E	
2DD0	DD	44	DC	54	83	00	01	DD	54	DC	44	93	52	10	2C	00	:47	
2DE0	11	DC	52	DD	44	DC	54	83	00	01	DD	54	CC	00	07	BD	:05	

2DFO	4E	DE	DC	54	83	00	00	10	2E	00	12	DC	44	83	00	01	:D3
Sum:	4C	DC	F7	0B	5E	A1	47	0A	19	DC	72	C5	2B	H6	2B	7E	:50
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2E00	DD	44	CC	00	01	DD	54	CC	00	07	BD	4E	DE	DC	44	34	:2F
2E10	06	17	02	BF	32	62	00	44	16	FF	3F	1E	02	10	CC	FF	:E7
2E20	FF	DD	44	CC	5D	DD	50	CC	00	1F	DD	46	CC	00	00	36	:91
2E30	06	CC	00	00	84	36	06	CC	02	7F	36	06	CC	00	02	36	:91
2E40	CC	00	01	36	06	CC	00	07	36	06	CC	00	02	34	06	37	:57
2E50	10	34	10	34	10	34	10	34	10	34	10	34	10	34	10	34	:2C
2E60	10	34	10	34	10	34	10	34	10	34	10	34	10	34	10	34	:76
2E70	CC	00	00	8D	4D	C1	36	06	CC	46	36	06	CC	00	20	BD	:E7
2E80	4D	C1	A4	CC	E4	C0	10	27	01	36	06	CC	46	82	00	1F	:58
2E90	26	00	5B	CC	54	83	00	00	10	2F	00	4D	CC	4A	C3	00	:AB
2EA0	01	DD	44	CC	5D	DD	44	CC	00	00	DD	60	CC	00	00	DD	:27
2EB0	42	CC	3C	36	06	CC	42	16	00	25	DE	50	34	06	17	01	:6D
2EC0	DF	32	62	0C	54	83	00	01	DD	54	DE	50	C3	00	01	34	:7E
2ED0	06	17	01	FF	32	62	DD	50	CC	42	C3	00	01	DD	42	A3	:82
2EE0	CA	2F	07	33	42	16	00	06	CC	00	07	BD	4E	DE	DC	46	:39
2EF0	B3	00	1E	10	26	00	5D	CC	4A	83	00	00	10	2F	00	4E	:6A
Sum:	82	5E	23	7C	AA	0E	73	75	A6	B1	5E	70	E7	AB	CA	F9	:95
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2F00	DC	4A	83	00	01	DD	4A	CC	00	00	DD	60	CC	54	D3	06	:E3
2F10	DD	54	CC	00	00	5D	CC	00	00	00	00	00	00	36	06	CC	:99
2F20	42	16	00	21	CC	FF	FF	CC	44	87	00	01	34	06	17	01	:39
2F30	A2	32	62	DD	44	CC	44	34	06	17	01	64	32	62	CC	42	:DF
2F40	C3	00	FF	FF	DD	42	A3	C4	2F	07	33	42	16	00	06	CC	:AB
2F50	07	BD	4E	DE	DC	46	83	00	1C	10	36	06	1A	DC	60	C3	:FA
2F60	00	01	DD	60	CC	00	93	CC	10	3F	00	05	CC	00	00	DD	:56
2F70	60	CC	46	83	00	10	26	00	1A	DC	60	83	00	01	DD	09	:59
2F80	60	CC	60	83	00	00	10	2C	00	04	CC	3C	DD	60	CC	60	:F0
2F90	36	06	CC	00	09	C3	00	01	34	06	37	10	34	10	17	00	:BE
2FA0	32	32	6A	BD	4D	D6	DD	46	CC	60	36	06	CC	00	09	C3	:DB
2FB0	00	01	34	06	37	10	34	10	17	00	18	32	6A	16	FE	AC	:5B
2FC0	CC	00	01	DD	42	CC	60	36	06	CC	42	16	00	15	DC	44	:CD
2FD0	C3	00	01	34	06	17	00	FF	32	62	DD	44	CC	42	C3	00	:A6
2FE0	01	DD	42	A3	C4	2F	E7	33	42	CC	4A	36	06	CC	06	HE	:13
2FF0	4D	1F	D3	60	93	4C	10	2D	00	05	CC	00	00	DD	46	CC	:7B
Sum:	6C	90	0C	3A	14	85	CB	BA	CF	F8	84	54	CB	6A	DE	4B	:5A
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3000	00	00	DD	42	CC	3C	36	06	CC	42	16	00	25	DC	42	36	:20
3010	06	CC	00	09	C3	00	01	36	06	CC	00	00	34	06	37	10	:2B
3020	34	10	37	10	34	10	17	03	8C	32	66	CC	42	C3	00	01	:EF
3030	DD	42	A3	C4	2F	07	33	42	17	0A	AE	16	00	05	CC	00	:AF
3040	DD	DD	46	CC	22	36	06	CC	24	36	06	CC	26	36	06	CC	:C0
3050	2B	34	06	37	10	34	10	37	10	34	10	37	10	34	10	17	:1A
3060	03	28	32	68	9E	16	58	49	EC	8B	DD	5C	CC	46	83	00	:72
3070	DD	10	26	00	CC	44	34	06	17	05	8B	32	62	16	00	37	:7C
3080	19	DC	22	36	06	CC	24	36	06	CC	5C	34	06	37	10	34	:7C
3090	10	37	10	34	10	17	03	1D	32	66	CC	CC	00	DD	46	39	:92
30A0	DC	54	83	00	00	10	2F	00	07	EC	62	DD	5C	16	00	05	:9B
30B0	CC	00	00	DD	5C	CC	42	36	06	CC	00	09	C3	00	01	36	:2E
30C0	06	CC	5C	34	06	37	10	34	10	37	10	34	10	17	02	05	:8C
30D0	32	66	39	EC	62	DD	62	CC	62	CC	AF	DD	5C	CC	5C	83	:00
30E0	7F	10	26	00	07	EC	62	C3	00	A2	DD	62	CC	62	DD	62	:8C
30F0	20	10	26	00	07	EC	62	C3	00	A2	DD	62	CC	62	DD	62	:8C
Sum:	04	33	F1	01	C6	4A	01	00	62	1A	4B	57	AB	17	30	29	:70
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3100	39	EC	62	36	06	CC	40	BD	4D	A9	36	06	EC	62	36	14	:42
3110	06	CC	00	5B	DD	4D	A1	CC	E4	C0	10	27	00	08	CC	C3	:EB
3120	FF	FF	DD	64	16	00	05	CC	00	00	64	EC	62	36	06	FF	:F1
3130	CC	00	60	BD	4D	A9	36	06	EC	62	36	06	CC	00	7B	BD	:A9
3140	4D	A1	A4	C0	E4	C0	10	27	00	08	CC	FF	DD	66	16	58	:58
3150	00	05	CC	00	00	DD	66	39	CC	26	83	00	00	10	26	00	:58
3160	0B	CC	80	BD	DD	42	16	00	05	CC	8F	8F	DD	42	CC	42	:35
3170	36	06	CC	1C	34	06	37	10	34	10	34	10	5A	32	6A	CC	:C2
3180	00	00	36	06	CC	00	00	36	06	CC	26	36	06	CC	2B	34	:8A
3190	06	37	10	34	10	37	10	34	10	37	10	34	10	17	01	ED	:AC
31A0	32	68	DD	6E	DC	26	83	00	00	10	26	00	24	CC	1E	DD	:9B
31B0	42	CC	00	00	36	06	CC	42	16	00	00	CC	CC	FF	FF	17	:6C
31C0	37	CC	42	C3	FF	FF	DD	42	A3	C4	2F	EF	73	42	16	00	:42
31D0	23	CC	54	D3	20	DD	54	CC	20	DD	42	CC	3C	36	06	CC	:C2
31E0	42	16	00	0A	17	00	11	CC	42	C3	00	01	DD	42	A3	C4	:F2
31F0	2F	F2	33	42	17	0B	C4	39	DC	6E	D3	42	DD	54	CC	CC	:9E
Sum:	DA	5A	57	9B	5E	EE	14	05	8B	92	C1	D2	7D	8B	DD	B7	:8F
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3200	00	DD	44	CC	00	09	36	06	CC	44	16	00	38	CC	DD	54	:9E
3210	16	58	49	EC	8B	83	00	00	10	27	00	1F	DC	42	36	06	:A1
3220	CC	44	36	06	CC	54	9E	16	58	49	EC	8B	34	06	37	10	:D9
3230	34	10	37	10	34	10	17	01	7C	32	66	CC	54	D3	1C	DD	:F7
3240	54	CC	44	C3	00	01	DD	44	A3	C4	2F	C1	33	42	39	EC	:A4
3250	62	34	06	BD	50	5A	32	62	EC	62	10	27	00	08	CC	00	:F0
3260	00	DD	54	16	00	05	CC	00	09	DD	54	CC	00	00	36	06	:5A
3270	CC	54	36	06	CC	26	36	06	CC	2B	34	06	37	10	34	10	:73
3280	37	10	34	10	37	10	34	10	17	01	02	32	68	DD	44	CC	:C7
3290	54	D3	28	9E	12	E6	8B	4F	83	00	00	10	27	00	09	CC	:5E
32A0	54	34	06	17	07	D1	32	62	CC	00	00	DD	52	CC	3C	36	:5A
32B0	06	CC	52	16	00	32	CC	44	D3	52	9E	16	58	49	EC	8B	:8D
32C0	DD	60	CC	60	10	27	00	19	CC	52	36	06	CC	54	36	06	:9F
32D0	CC	60	34	06	37	10	34	10	37	10	34	10	17	00	D6	32	:AB
32E0	66	CC	52	C3	00	01	DD	52	A3	C4	2F	CA	33	42	39	17	:AC
32F0	0A	4A	17	07	EC	CC	00	00	DD	42	CC	00	09	36	06	CC	:36
Sum:	C6	A3	F8	75	4A	73	DA	49	00	CC	34	55	71	1F	CC	37	:E1

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3300	42	16	00	80	CC	00	00	36	06	CC	42	36	06	CC	26	36	:72
3310	06	CC	2B	34	06	37	10	34	10	37	10	34	10	37	10	34	:D5
3320	10	17	00	69	32	68	DD	54	CC	42	D3	28	9E	12	E6	8B	:95
3330	4F	83	00	00	10	27	00	09	CC	42	34	06	17	07	38	3C	:F2
3340	62	CC	00	00	DD	44	DC	3C	36	06	CC	44	16	00	28	CC	:62
3350	44	36	06	00	DD	42	36	06	CC	54	D4	4C	9E	16	58	49	:ED
3360	BB	34	06	37	10	34	10	37	10	34	10	17	00	47	32	66	:D1
3370	DC	44	CC	00	01	DD	44	A3	C4	2F	D4	33	42	CC	42	C3	:C5
3380	00	01	DD	42	A3	C4	10	2F	FF	76	33	42	39	EC	62	36	:E7
3390	05	EC	64	26	06	EC	68	E3	C1	36	06	CC	1C	BD	4D	1F	:71
33A0	E3	C1	DD	60	EC	68	10	27	00	08	CC	60	D3	3E	DD	60	:FA
33B0	00	60	DD	60	59	EC	68	26	06	EC	62	C3	00	01	36	06	:8E
33C0	EC	64	34	06	37	10	34	10	37	10	34	10	BD	50	00	32	:DF
33D0	66	39	CC	00	00	DD	EC	62	36	06	CC	00	41	BD	4D	:49	
33E0	B9	36	06	EC	62	36	06	CC	00	49	BD	4D	B9	AA	CC	EA	:B8
33F0	00	36	06	EC	62	36	06	CC	00	55	BD	4D	B9	AA	CC	EA	:B8
Sum:	44	1D	FE	46	0D	AC	B1	EC	8B	59	8B	7B	90	74	38	26	:14
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3400	CC	36	06	EC	62	36	06	CC	00	45	BD	4D	B9	AA	CC	EA	:AB
3410	00	36	06	EC	62	36	06	CC	00	4F	BD	4D	B9	AA	CC	EA	:8E
3420	00	10	27	00	05	CC	FF	FF	DD	60	EC	62	36	06	CC	00	:59
3430	61	BD	4D	89	36	06	EC	62	36	06	CC	00	69	BD	4D	B9	:E2
3440	0A	00	1F	CC	10	06	EC	62	36	06	CC	00	75	BD	4D	B9	:E2
3450	AA	CC	EA	CC	26	06	06	EC	62	36	06	CC	00	65	BD	4D	:B9
3460	AA	CC	EA	CC	26	06	06	EC	62	36	06	CC	00	6F	BD	4D	:DB
3470	AA	CC	EA	CC	10	27	00	05	CC	FF	FF	DD	60	CC	00	1F	:76
3480	69	39	DD	46	34	06	17	FF	4F	32	62	10	27	00	0B	CC	:12
3490	00	20	DD	42	17	00	D3	16	00	1F	DC	46	BD	42	CC	00	:12
34A0	00	DD	54	CC	42	DD	46	CC	46	34	06	17	05	AA	32	62	:E8
34B0	BD	4D	06	DD	42	CC	46	34	06	17	05	9E	32	62	CC	4E	:C7
34C0	34	06	17	FF	DD	72	62	36	06	CC	46	36	06	CC	00	4E	:A5
34D0	BD	4D	B9	AA	CC	EA	CC	26	06	CC	46	36	06	CC	00	6E	:AB
34E0	BD	4D	B9	AA	CC	EA	CC	26	06	CC	42	36	06	CC	00	20	:59
34F0	BD	4D	B9	AA	CC	EA	CC	10	27	00	05	CC	FF	FF	DD	54	:02
Sum:	D1	A9	4D	C9	CD	20	D3	FB	4F	EB	B1	4A	06	D5	A2	79	:36
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3500	DD	42	34	06	17	FE	CC	32	62	34	06	CC	42	36	06	CC	:3E
3510	00	BD	4D	B9	AA	CC	EA	CC	36	06	CC	54	AA	CC	EA	CC	:57
3520	36	06	CC	42	36	06	CC	02	BD	4D	B9	AA	CC	EA	CC	10	:58
3530	27	FF	70	DD	36	97	00	20	10	26	00	CC	DC	46	DD	54	:F0
3540	CC	02	DD	46	16	00	03	17	00	01	39	DC	46	34	06	17	:80
3550	CC	02	DD	46	16	00	12	CC	42	93	00	10	26	00	07	CC	:F3
3560	00	DD	46	16	00	03	17	00	01	39	DC	46	34	06	17	FF	:FB
3570	90	32	62	CC	68	10	27	00	07	CC	46	83	00	20	DD	46	:84
3580	CC	42	34	06	17	FF	7A	32	62	CC	66	10	27	00	07	CC	:DC
3590	42	83	00	20	DD	42	CC	46	32	62	CC	01	00	DD	4D	1F	:58
35A0	D3	42	34	06	BD	4A	D7	32	62	DD	46	CC	46	4F	DD	42	:74
35B0	CC	46	1F	89	4F	DD	4D	CC	32	83	00	00	10	26	00	43	:46
35C0	CC	46	36	06	CC	01	00	DD	4F	93	36	34	06	BD	48	:5C	
35D0	93	32	62	CC	46	CC	42	83	00	DE	10	26	00	07	CC	46	:18
35E0	C7	00	01	DD	46	CC	42	83	00	DF	10	26	00	07	CC	46	:C6
35F0	C3	00	02	CC	46	CC	46	34	06	17	00	08	32	62	CC	00	:C3
Sum:	33	1C	51	AD	33	A7	FB	AF	0A	CE	7B	3B	D5	1A	FA	7B	:BD
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3600	00	DD	46	39	CC	22	36	06	CC	24	36	06	CC	26	36	06	:10
3610	DD	28	34	06	37	10	34	10	37	10	34	10	37	10	34	10	:DF
3620	17	FD	BA	32	68	DD	42	CC	3B	10	27	00	BE	CC	36	06	:8E
3630	06	CC	10	27	00	06	CC	00	01	36	06	CC	28	34	06	37	:10
3640	34	10	37	10	34	10	37	10	34	10	17	FD	B3	42	68	DD	:25
3650	44	CC	44	9E	16	58	49	EC	8B	03	00	10	27	00	09	:F3	
3660	CC	00	07	BD	4E	DE	16	00	CC	DC	44	93	42	10	27	00	:80
3670	7A	CC	44	83	00	01	DD	52	CC	42	36	06	CC	52	16	00	:EB
3680	21	CC	FF	FF	CC	52	C3	00	01	58	49	9E	16	30	8E	76	:22
3690	10	CC	52	9E	16	58	49	EC	8B	ED	D1	CC	52	C3	FF	FF	:B0
36A0	DD	52	A3	C4	2C	DD	33	42	CC	3C	DD	52	CC	22	C3	00	:1A
36B0	01	36	06	CC	52	16	00	2D	CC	FF	FF	CC	52	36	06	CC	:5B
36C0	24	36	06	CC	42	D3	52	93	22	9E	16	58	49	EC	8B	74	:58
36D0	06	37	10	34	10	37	10	34	10	17	FD	D9	32	66	CC	52	:CE
36E0	C3	FF	FF	DD	52	A3	C4	2C	DF	C3	42	CC	42	58	49	9E	:24
36F0	16	30	8B	36	10	EC	62	ED	D1	CC	22	36	06	CC	24	36	:93
Sum:	C9	72	68	F5	3D	56	E6	7C	A4	3F	6A	BF	CC	A4	A9	AD	:5F
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3700	06	EC	62	34	06	37	10	34	10	37	10	34	10	17	FC	A5	:5C
3710	32	66	CC	2E	83	FF	FF	10	26	00	07	CC	22	C3	00	01	:22
3720	DD	22	CC	22	36	06	CC	3C	BD	40	49	36	06	CC	26	AA	:E6
3730	CC	E4	00	10	27	00	07	DC	24	C3	00	01	DD	24	17	00	:7E
3740	7D	39	CC	24	DD	52	CC	00	09	36	06	CC	52	16	00	67	:A1
3750	CC	00	00	36	06	CC	52	36	06	CC	26	36	06	CC	2B	34	:D8
3760	06	37	10	34	10	37	10	34	10	37	10	34	10	17	FC	1D	:D7
3770	32	68	DD	42	CC	00	00	DD	44	DC	3C	36	06	CC	44	16	:30
3780	00	28	DC	44	36	06	CC	52	36	06	CC	42	D3	44	9E	16	:D7
3790	58	49	EC	8B	34	06	37	10	34	10	37	10	34	10	17	FC	:7B
37A0	14	32	66	CC	44	CC	00	01	DD	44	A3	C4	2F	D4	33	42	:90
37B0	CC	52	C3	00	01	DD	52	A3	C4	2F	95	33	42	39	DC	22	:FB
37C0	93	3C	10	2F	DD	20	CC	26	10	27	00	08	CC	00	00	DD	:18
37D0	22	16	00	04	CC	20	DD	22	CC	26	34	06	17	F5	01	32	:B2
37E0	62	DD	26	17	F9	72	CC	22	83	00	00	10	2C	00	1F	CC	:9F
37F0	26	10	27	00	07	CC	1E	DD	22	16	00	04	DC	3C	DD	22	:8E
Sum:	64	F1	59	30	DB	38	F0	16	5B	B7	2E	E6	51	62	9B	A3	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3800	CC	26	34	06	17	F4	D9	32	62	DD	26	17	F9	4A	CC	00	:DD
3810	00	DD	5C	DD	24	B3	00	09	10	2F	00	51	CC	CC	00	01	:DD
3820	42	DC	24	B3	00	09	36	06	CC	42	16	00	2C	00	DC	2B	:C3



3830	00	01	DD	28	DC	28	93	14	10	2E	00	12	CC	FF	FF	DD	:A8
3840	5C	CC	00	00	34	06	17	FA	06	32	62	16	00	04	DC	14	:17
3850	DD	2B	DC	42	C3	00	01	DD	42	A3	C4	2F	DD	33	42	DC	:B0
3860	5C	10	27	00	03	17	02	79	CC	00	09	DD	24	DC	24	B3	:81
3870	00	00	10	2C	00	53	CC	FF	FF	DD	42	DC	24	36	06	DC	:90
3880	42	16	00	31	CC	FF	FF	DC	28	B3	00	01	DD	28	DC	28	:E4
3890	83	00	00	10	20	00	12	CC	FF	FF	DD	5C	CC	FF	FF	C4	:D3
38A0	06	17	F9	AD	32	62	16	00	05	CC	00	00	DD	28	DC	42	:5F
38B0	C3	FF	FF	AD	42	A3	C4	2C	CB	33	42	DC	5C	10	27	00	:22
38C0	C3	17	02	1D	CC	00	00	DD	24	39	DC	22	36	06	DC	24	:79
38D0	36	06	DC	26	36	06	DC	28	34	06	37	10	34	10	37	10	:8A
38E0	34	10	37	10	34	10	17	FA	A4	32	68	DD	42	DC	3C	36	:8B
38F0	06	DC	24	36	06	CC	00	01	36	06	DC	28	34	06	37	10	:D0

Sum: B4 19 D5 4D BA FE 66 78 9A 26 23 EB 97 C5 A0 E4 :30

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3900	34	10	37	10	34	10	37	10	34	10	17	FA	80	32	68	DD	:62
3910	44	DC	42	DD	52	DC	44	83	00	01	36	06	DC	52	16	00	:B5
3920	1E	DC	52	58	49	9E	16	30	8B	36	10	DC	52	C3	00	01	:94
3930	9E	16	58	49	EC	8B	36	10	DC	52	C3	00	01	DD	52	A3	:E4
3940	C4	2F	DD	33	42	DC	44	58	49	9E	16	30	8B	36	10	DC	:88
3950	00	00	ED	D1	CC	00	00	DD	52	DC	C3	93	22	36	06	DC	:9E
3960	52	16	00	2A	DC	22	D3	52	36	06	DC	24	36	06	DC	42	:48
3970	D3	52	9E	16	58	49	EC	8B	36	06	DC	34	06	37	10	34	:FD
3980	34	10	17	FA	30	32	66	DC	52	C3	00	01	DD	52	A3	C4	:A5
3990	2F	D2	33	42	39	CC	02	58	36	06	CC	00	B6	36	06	CC	:9B
39A0	00	2A	36	06	CC	00	04	34	06	37	10	34	10	37	10	34	:74
39B0	10	37	10	34	10	17	03	A8	32	68	39	CC	02	62	36	06	:9C
39C0	CC	00	B6	36	06	CC	00	2A	36	06	CC	00	04	34	06	37	:31
39D0	10	34	10	37	10	34	10	37	10	34	10	17	03	82	32	68	:A0
39E0	39	CC	02	76	36	06	CC	00	B6	36	06	CC	00	2A	36	06	:A9
39F0	CC	00	04	34	06	37	10	34	10	37	10	34	10	37	10	34	:9B

Sum: 71 B8 E8 5F 94 AE DC 4B 6C 2E 8C EB 82 DE 66 1E :CE

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3A00	10	17	03	5C	32	68	39	CC	02	62	36	06	CC	00	B6	36	:87
3A10	06	CC	00	2A	36	06	CC	00	04	34	06	37	10	34	10	37	:04
3A20	10	34	10	37	10	34	10	17	03	36	32	68	39	CC	00	38	:09
3A30	36	06	CC	00	B6	36	06	CC	00	36	06	CC	00	04	34	06	:8B
3A40	37	10	34	10	37	10	34	10	17	03	11	32	FE				:FE
3A50	68	39	CC	00	52	36	06	CC	00	B6	36	06	CC	00	62	36	:43
3A60	CC	00	04	34	06	37	10	34	10	37	10	34	10	37	10	34	:9B
3A70	10	17	02	EC	32	68	39	CC	00	04	DD	00	CC	00	00	36	:07
3A80	06	EC	62	36	06	CC	00	12	8D	4F	1F	36	06	DC	28	36	:0D
3A90	06	EC	62	E3	C1	9E	12	6E	8B	4F	C3	00	30	36	06	CC	:63
3AA0	00	04	34	06	37	10	34	10	37	10	34	10	37	10	34	10	:DF
3AB0	17	02	AD	32	68	CC	00	07	DD	40	39	CC	00	34	36	06	:C5
3AC0	CC	00	B6	36	06	CC	00	2A	36	06	CC	00	04	34	06	37	:31
3AD0	10	34	10	37	10	34	10	37	10	34	10	17	02	82	32	68	:9F
3AE0	39	CC	00	32	36	06	CC	00	B5	36	06	CC	00	5B	36	06	:93
3AF0	CC	00	BF	36	06	CC	00	00	36	06	CC	00	07	36	06	CC	:A2

Sum: DB 5B 0F 13 A7 D5 C0 0B 49 6F CB FO 6E 3D 5D D9 :F0

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3B00	00	01	34	06	37	10	34	10	37	10	34	10	37	10	34	10	:DC
3B10	37	10	34	10	37	10	34	10	37	10	34	10	17	02	7E	32	:6A
3B20	6E	CC	02	57	36	06	CC	00	B5	36	06	CC	02	7F	36	06	:15
3B30	CC	00	BF	36	06	CC	00	36	06	CC	00	07	36	06	CC	:AA	
3B40	00	01	34	06	37	10	34	10	37	10	34	10	37	10	34	10	:DC
3B50	37	10	34	10	37	10	34	10	37	10	34	10	17	02	3E	32	:2A
3B60	6E	CC	02	61	36	06	CC	00	B5	36	06	CC	02	61	36	06	:01
3B70	CC	00	BF	36	06	CC	00	36	06	CC	00	07	36	06	CC	:AA	
3B80	00	00	34	06	37	10	34	10	37	10	34	10	37	10	34	10	:DB
3B90	37	10	34	10	37	10	34	10	37	10	34	10	17	01	FE	32	:E9
3BA0	6E	CC	02	6B	36	06	CC	00	B5	36	06	CC	02	6B	36	06	:15
3BB0	CC	00	BF	36	06	CC	00	36	06	CC	00	07	36	06	CC	:AA	
3BC0	00	00	34	06	37	10	34	10	37	10	34	10	37	10	34	10	:DB
3BD0	37	10	34	10	37	10	34	10	37	10	34	10	17	01	BE	32	:A9
3BE0	6E	CC	02	75	36	06	CC	00	B5	36	06	CC	02	75	36	06	:29
3BF0	CC	00	BF	36	06	CC	00	36	06	CC	00	07	36	06	CC	:AA	

Sum: C4 72 A4 CB AB CB D0 80 64 70 E8 B0 5C DE 3B 50 :90

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3C00	00	00	34	06	37	10	34	10	37	10	34	10	37	10	34	10	:DB
3C10	37	10	34	10	37	10	34	10	37	10	34	10	17	01	7E	32	:6A
3C20	4E	CC	02	57	36	06	CC	00	B5	36	06	CC	02	7F	36	06	:15
3C30	17	FD	62	DC	36	10	27	00	03	17	FD	7F	DC	5A	10	27	:A2
3C40	00	03	17	FD	9C	DC	32	10	27	00	03	17	FD	B9	DC	2C	:D0
3C50	10	27	00	03	17	FD	6B	39	CC	22	58	49	58	49	58	49	:3E
3C60	58	49	C3	00	10	36	06	CC	00	24	36	06	CC	00	12	BD	:4D
3C70	1F	C3	00	1F	36	06	CC	00	22	58	49	58	49	58	49	58	:AF
3C80	C3	00	1F	36	06	CC	00	24	36	06	CC	00	12	BD	4D	1F	:C3
3C90	00	11	36	06	CC	00	04	36	06	CC	00	07	36	06	CC	00	:34
3CA0	01	34	06	37	10	34	10	37	10	34	10	37	10	34	10	37	:13
3CB0	10	34	10	37	10	34	10	37	10	34	10	17	00	DF	32	6E	:03
3CC0	39	DC	22	36	06	DC	24	34	06	37	10	34	10	37	10	03	:52
3CD0	32	64	39	EC	62	58	49	58	49	58	49	58	49	C3	00	10	:78
3CE0	36	06	EC	64	36	06	CC	00	12	BD	4D	1F	C3	00	10	36	:D4
3CF0	06	EC	62	58	49	58	49	58	49	58	49	C3	00	1F	36	06	:F6

Sum: BE CA DE 99 9D 1B 46 3C CA 0D 09 1D 06 2E 7E 2E :10

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3D00	EC	64	36	06	CC	00	12	BD	4D	1F	C3	00	10	36	06	CC	:6E
3D10	00	04	36	06	CC	00	07	36	06	CC	00	00	34	06	37	10	:9C
3D20	34	10	37	10	34	10	37	10	34	10	37	10	34	10	37	10	:2C
3D30	34	10	37	10	34	10	17	00	64	32	6E	39	17	00	8F	CC	:95
3D40	FC	82	DD	70	9E	70	CC	00	0D	E7	80	CC	00	00	E7	80	:4C
3D50	CC	00	00	E7	80	CC	00	00	E7	80	9F	70	17	00	80	39	:4C
3D60	17	00	6B	CC	FC	82	DD	70	9E	70	CC	00	19	E7	80	CC	:4F

3D70	40	E7	80	EC	68	E7	80	CC	00	00	E7	80	CC	00	01	E7	:49
3D80	80	CC	00	01	E7	80	EC	62	ED	81	EC	64	ED	81	CC	00	:FA
3D90	01	E7	80	EC	66	E7	80	9F	70	17	00	43	39	17	00	2E	:08
3DA0	CC	FC	B2	D0	70	9E	70	CC	00	15	E7	80	EC	6C	E7	80	:AC
3DB0	EC	6A	E7	80	EC	62	ED	81	EC	64	ED	81	EC	66	ED	81	:F7
3DC0	EC	68	ED	81	EC	6E	E7	80	9F	70	17	00	12	39	B6	FD	:A7
3DD0	05	2B	F8	CC	00	00	F7	FD	05	B6	FD	05	2A	FB	39	CC	:52
3DE0	00	00	F7	FD	05	39	C6	00	1F	9B	7E	4F	F0	00	00	00	:6F
3DF0	6E	9F	FB	FA	00	00	00	00	00	00	00	00	00	00	00	00	:02



42B0	11	45	35	C4	20	17	45	46	C4	B3	7F	45	5D	C4	B7	01	E5	
42C0	46	3E	C4	B8	0B	46	3F	C4	C1	02	46	4A	C4	C2	02	46	75	
42D0	4C	D4	C4	02	46	4E	C4	CB	01	46	50	C4	CF	01	46	51	BB	
42E0	C4	D7	01	46	53	C4	D8	01	46	53	C4	D8	01	46	54	C4	68	
42F0	DD	0B	46	55	C5	20	02	46	60	C5	B2	01	46	62	C5	B6	AR	
Sum:	34	8C	79	F1	92	BD	D4	9C	65	21	5A	7E	66	9C	2B	79	F3	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
4300	01	46	63	C5	B7	02	46	64	C5	BF	01	46	65	C5	D0	01	89	
4310	46	67	C5	C2	01	46	68	C5	CD	01	46	69	C5	D7	01	46	08	
4320	6A	C5	DA	01	46	68	C5	DC	02	46	6C	C5	DD	05	46	6E	68	
4330	C6	20	04	46	73	C6	B5	01	46	77	C6	B7	01	46	78	C6	DE	
4340	B8	01	46	79	C6	BC	02	46	7A	C6	C1	01	46	7C	C6	D5	A1	
4350	02	46	7D	C6	D6	02	47	21	C6	D7	01	47	23	C6	DD	04	7A	
4360	47	24	C7	DA	01	47	28	C8	20	02	47	29	C8	B2	01	47	98	
4370	2B	CB	B7	01	47	2C	CB	BA	01	47	2D	CB	C2	01	47	2E	15	
4380	C8	DD	06	47	2F	C9	20	04	47	35	C9	B3	08	47	39	C9	57	
4390	BF	01	47	41	C9	DD	01	47	42	CA	20	0D	47	43	CA	B2	68	
43A0	1B	47	50	CA	B4	01	47	68	CA	B6	01	47	69	CA	B7	02	91	
43B0	47	6A	CA	B8	13	47	6C	CA	B8	02	48	21	CA	B8	01	48	B6	
43C0	23	CA	BC	02	48	24	CA	BD	01	48	26	CA	BE	01	48	27	05	
43D0	CA	C0	04	48	28	CA	C1	02	48	2C	CA	C2	09	48	2E	CA	D4	
43E0	C4	01	48	37	CA	C5	02	48	38	CA	CF	01	48	3A	CA	D4	0F	
43F0	01	48	3B	CA	DD	1F	48	3C	CB	20	25	48	3B	CB	B2	01	FF	
Sum:	3B	27	F1	3D	2B	5D	0A	AF	94	7B	C5	61	EB	39	17	54	8F	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
4400	47	22	CB	B4	01	49	23	CB	B7	02	49	24	CB	B9	01	49	16	
4410	26	CB	BA	01	49	27	CB	BB	01	49	28	CB	BC	02	49	29	0F	
4420	CB	C2	05	49	2B	CB	C9	01	49	30	CB	D2	02	49	31	CB	F8	
4430	D3	01	49	33	CB	D4	01	49	34	CB	D5	01	49	35	CB	D6	2D	
4440	11	49	36	CB	D9	02	49	37	CB	DA	01	49	38	CB	DB	0A	B0	
4450	49	4A	CC	20	21	49	54	CC	B3	03	49	75	CC	B7	02	49	4B	
4460	7B	CC	B8	09	49	7A	CC	C1	01	4A	25	CC	C2	05	4A	26	C8	
4470	CC	C5	01	4A	2B	CC	DE	4A	2C	CC	B2	0E	4A	3A	CC	D2	12	
4480	B7	04	4A	4B	CD	C2	03	4A	4C	CD	D7	01	4A	4F	CD	DD	5D	
4490	DD	0A	50	CE	20	12	4A	5D	CE	B3	39	4A	4F	CE	B4	01	44	
44A0	4B	4A	CE	B5	01	4B	4B	CE	B8	09	4B	4C	CE	D0	01	4B	AF	
44B0	55	CE	C2	02	4B	56	CE	C4	01	4B	58	CE	DB	01	4B	59	09	
44C0	CE	DB	01	4B	5A	CE	DD	05	4B	5B	CF	20	04	4B	60	CF	12	
44D0	B2	06	4B	64	CF	B7	01	4B	6A	CF	BB	04	4B	6B	CF	BB	6E	
44E0	01	4B	6F	CF	DD	02	4B	70	CF	C0	03	4B	72	CF	C2	03	E7	
44F0	4B	75	CF	C3	01	4B	7B	CF	C7	01	4B	79	CF	D5	01	4B	69	
Sum:	DB	DB	42	7D	CE	E7	05	7A	24	5B	D5	4B	A6	42	6B	B3	4B	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
4500	7A	CF	DB	01	4B	7B	CF	DD	05	4B	7C	DD	20	05	4C	23	C7	
4510	DD	BB	01	4C	2B	DD	C2	02	4C	2D	DD	C5	01	4C	2B	DD	E6	
4520	C9	02	4C	2C	DD	04	01	4C	2E	DD	D6	01	4C	2F	DD	D8	2C	
4530	01	4C	30	DD	DD	02	4C	31	D1	20	47	0C	4C	33	D1	BB	01	AA
4540	4C	3A	D1	BA	01	4C	3B	D1	BD	01	4C	3C	DD	B2	09	4C	B9	
4550	3D	D2	BD	01	4C	46	D2	C2	01	4C	47	DD	D2	DD	06	4C	4B	DD
4560	D3	20	03	4C	4E	D3	B3	09	4C	51	D3	BB	04	4C	5A	D3	C4	
4570	C1	02	4C	5E	D3	C2	01	4C	60	D3	C4	01	4C	61	D3	DD	97	
4580	01	4C	62	D3	D7	01	4C	63	D3	DD	DD	05	4C	64	DA	20	0B	6A
4590	4C	69	DA	B8	06	4C	71	DA	BD	01	4C	72	DA	D7	C5	01	4C	3F
45A0	7B	DA	CC	01	4C	79	DA	DD	01	4C	7A	DD	20	06	4C	7B	13	
45B0	D5	B2	01	4D	23	D5	B3	19	4D	24	DA	DD	06	4D	3D	DA	66	
45C0	B3	1A	4D	43	DA	B8	07	4D	5D	DA	C4	01	4D	64	D7	DD	DF	
45D0	03	4D	65	DF	B3	04	4D	6B	DF	BB	04	4D	6C	DF	DD	DD	FF	
45E0	4D	70	DD	20	0C	4D	7B	DD	BB	01	4E	2B	DD	C2	04	4E	77	
45F0	27	DD	D4	02	4E	2B	DD	D5	DA	01	4E	2D	DD	D6	1A	4E	77	
Sum:	F5	F0	94	C3	BE	17	B7	CE	BE	00	37	4D	64	B9	31	55	7B	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
4600	DD	DD	DD	4E	51	D9	20	01	4E	5C	D9	B2	04	4E	5D	DA	17	
4610	B2	10	4E	61	DA	B7	02	4E	71	DA	C2	04	4E	73	DA	DD	82	
4620	0B	4E	77	DD	20	0F	4E	7A	DD	B2	10	4F	DD	DD	BB	05	F5	
4630	4E	7B	DD	01	4F	4E	DD	20	03	4F	01	4D	DD	B2	02	4F	40	
4640	4A	DD	B7	01	4F	4E	DD	B8	03	4F	47	DD	DD	01	4F	4F	CA	
4650	DD	C0	02	4F	4A	DD	C6	04	4F	4C	DD	CB	01	4F	4D	DD	95	
4660	D7	02	4F	4E	DD	04	4F	5D	DD	B8	F7	1B	26	21	4E	DD	0F	
4670	21	27	DD	B7	1B	36	00	47	3E	F9	33	7C	32	EB	DA	AD	AD	
4680	21	32	DD	00	34	77	BA	00	47	3E	F9	33	7C	32	EB	DA	AD	
4690	EF	09	49	33	44	10	EF	0C	49	33	46	49	3F	10	E6	4E	30	F7
46A0	1E	89	ED	8C	33	5F	E7	5E	BD	37	66	4F	84	02	27	12	8F	
46B0	86	04	30	8C	1B	DA	3E	C6	01	E7	5E	BD	37	66	4F	84	02	27
46C0	35	F7	86	03	30	8C	0C	20	EC	16	00	00	00	42	45	1B	F7	
46D0	41	0B	0A	1B	4A	01	1B	4C	00	00	00	00	00	00	00	01	99	
46E0	34	1A	E7	5C	30	8C	F3	BD	30	F3	6E	8C	07	27	03	BD	B4	
46F0	2D	F0	35	94	00	34	1B	E6	80	BD	E5	4A	2E	F9	35	94	4C	
Sum:	97	79	E6	3B	E7	BD	47	57	CB	17	8B	A9	C4	1B	D2	34	6B	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
4700	39	10	AE	4B	C6	DD	0B	86	4A	50	8D	BD	0E	4B	66	A6	A2	
4710	4E	AD	5D	27	EB	30	8C	B1	EC	01	10	8C	2B	2A	27	0C	70	
4720	ED	04	BD	3D	F0	4F	BD	24	86	01	9D	20	6F	8C	C5	F5	2E	
4730	A6	4D	27	05	BD	AA	4A	20	F9	EC	7E	10	83	27	2A	27	E9	
4740	04	6A	5D	20	CE	86	FF	A7	8C	AA	20	F5	33	F5	34	42	38	
4750	33	41	C4	0B	30	54	6F	80	5A	26	F9	30	1B	46	4F	C6	33	
4760	0B	E7	5F	4E	25	3B	46	24	20	E6	5E	5B	58	5B	58	EB	0A	
4770	E4	33	C8	DA	33	C5	AE	C1	17	00	A3	26	F9	35	42	A6	08	
4780	C0	30	54	85	0B	17	FF	6D	39	A6	5E	4B	AB	E4	33	C6	62	
4790	33	C8	DA	A6	C4	33	44	17	00	84	F7	20	DF	30	8D	24	24	
47A0	00	00	00	EC	B1	27	09	10	A3	3E	27	32	A6	80	20	F3	E6	A6
47B0	4F	AE	E4	30	54	48	48	48	48	C4	02	27	34	4C	A0	5E	E8	
47C0	33	C6	33	CB	DA	86	08	34	02	A6	C1	C6	0B	48	66	84	F3	F3
47D70																		

47F0	75	E6	5E	33	C6	33	CB	DA	B6	0B	34	06	EC	C1	34	02	2C
Sum:	89	3D	1B	3A	E6	AB	F7	5D	A3	E6	25	0A	EO	1B	43	14	0A
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4800	A6	32	35	02	25	02	58	49	58	49	66	84	58	49	6A	E4	285
4810	26	F8	A6	80	35	06	5A	F8	01	26	DD	16	FF	5F	C6	08	:82
4820	48	34	02	A6	F8	04	35	02	2F	14	69	80	5A	26	F1	30	:24
4830	18	A6	F8	03	2F	04	6A	F8	03	39	6C	F8	03	39	66	80	:10
4840	20	EA	21	27	00	21	4A	00	21	4B	00	21	4C	00	21	4D	:04
4850	00	21	4E	00	21	4F	00	21	50	00	21	51	00	21	52	00	:35
4860	21	53	00	21	54	00	21	55	00	21	56	00	21	57	00	21	:6F
4870	58	00	21	59	00	21	5A	00	21	5B	00	21	22	01	21	23	:51
4880	01	00	24	11	EC	05	C1	00	26	31	6F	8C	68	6F	8C	00	:0D
4890	64	81	30	25	14	81	3A	25	4C	81	41	25	0C	81	5B	25	:6E
48A0	44	81	41	25	04	81	7B	25	3C	30	8C	5A	C6	9D	A1	80	:40
48B0	27	3F	A1	81	5A	26	F7	0C	00	30	35	91	4F	C1	DE	27	:A6
48C0	15	4C	C1	DF	27	04	4C	C1	01	27	0B	4C	C1	02	27	06	:B4
48D0	C1	20	27	F7	20	E1	30	8C	1F	48	30	8A	EC	84	ED	8C	:82
48E0	14	A6	85	00	20	AF	1F	89	8A	00	C3	23	00	E3	9C	06	:20
48F0	09	EC	84	20	F7	00	00	00	00	01	00	02	01	00	02	E0	:36
Sum:	4B	CF	68	A1	64	C5	5C	5B	C5	8D	20	F2	B1	D9	80	17	:35
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4900	21	21	2A	DE	21	2B	22	21	4B	3F	21	29	3B	21	2B	3A	:68
4910	21	2F	DE	21	2C	A1	21	23	2C	21	24	2E	21	25	27	21	:86
4920	46	5E	21	30	7E	21	31	80	21	3C	2F	21	3F	2B	21	4A	:F0
4930	29	21	4B	5B	21	4C	5D	21	4D	A5	21	26	23	21	74	24	:F4
4940	21	70	25	21	73	26	21	75	8A	21	7A	2B	21	5C	2D	21	:91
4950	5D	89	21	5E	3C	21	63	3E	21	64	60	21	6C	3D	21	61	:24
4960	8C	21	62	8B	21	66	87	21	65	40	21	77	5C	21	6F	7B	:6A
4970	21	50	7D	21	51	7C	21	43	4E	21	56	A3	21	57	6D	21	:B2
4980	7B	EC	21	7C	F9	22	29	2B	22	23	8B	21	7F	2A	21	5F	:E5
4990	8D	21	79	9B	22	2A	99	22	28	21	22	2C	E2	22	2D	F1	:F2
49A0	21	6E	F3	22	2B	E0	21	7D	97	21	60	94	21	76	F0	21	:9E
49B0	7B	9D	21	72	F7	21	71	FB	21	41	FC	21	44	FC	21	45	:47
49C0	F4	21	47	FA	21	49	EB	21	69	8E	21	6A	93	22	21	8F	:B3
49D0	22	25	92	22	27	9A	21	5A	90	21	5B	E9	21	39	F5	21	:9C
49E0	4E	F6	21	47	EB	21	38	EE	22	24	EF	22	22	E1	21	7E	:0C
49F0	E3	22	26	FD	21	54	F8	21	55	95	21	5B	9A	21	59	EA	:13
Sum:	C4	A7	68	C2	98	07	8D	EF	09	A5	7B	D3	E2	B1	7D	B5	:6D
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4A00	21	3A	00	21	42	E4	24	70	E5	25	70	E6	24	71	E7	25	:B7
4A10	17	9C	25	74	9E	21	2B	9F	21	6D	86	21	2D	85	21	2E	:65
4A20	81	21	68	82	21	7B	83	21	6C	84	21	6D	81	24	22	82	:F0
4A30	24	24	83	24	76	84	24	2B	85	24	2A	86	24	2B	87	24	:28
4A40	2D	B8	24	2F	B9	24	31	8A	24	33	8B	24	35	8C	24	37	:B2
4A50	8D	24	39	BE	24	3B	8F	24	3D	00	24	3F	C1	24	41	02	:62
4A60	24	44	C3	24	46	C4	24	4A	C5	24	4A	C6	24	4B	C7	24	:18
4A70	4C	C8	24	4D	C9	24	4E	CA	C8	24	4F	CB	24	52	CC	24	:83
4AB0	CD	24	58	CC	24	5B	CF	24	5E	D0	24	5F	D1	24	60	D2	:61
4A90	24	61	D3	24	62	DA	24	64	D5	24	66	D6	24	68	D7	24	:F6
4AA0	69	D8	24	6A	D9	24	6B	DA	24	6C	DB	24	6D	DC	24	6F	:7C
4AB0	A6	24	72	DD	24	73	5F	21	32	AA	21	22	6F	21	24	A8	:0D
4AC0	24	23	A9	24	25	AA	24	27	AB	24	29	AC	24	23	AD	24	:2A
4AD0	65	AE	24	67	AF	24	43	34	11	86	00	47	BD	00	80	A7	:0A
4AE0	8D	00	AE	EC	05	C1	20	26	1E	C6	05	30	8C	7D	A1	80	:D6
4AF0	27	05	EA	26	F9	20	2C	EB	8D	00	95	EB	8D	00	90	A6	:AC
Sum:	CE	5A	9A	6F	C8	ED	CB	37	61	14	7E	60	65	AB	3B	99	:19
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4B00	8D	00	8E	1E	89	20	38	30	8C	66	C6	06	A1	80	27	36	:86
4B10	5A	26	F9	30	8C	60	6C	05	A1	80	27	25	5A	26	F9	81	:C7
4B20	50	27	32	C6	06	EA	67	8C	6A	30	8C	50	EC	65	10	A3	:81
4B30	27	31	6A	8C	5D	26	F6	10	83	57	4F	27	13	4F	5F	35	:1D
4B40	91	86	DE	A7	8C	4B	86	05	3C	E7	8C	43	A6	66	20	99	:86
4B50	CC	A6	00	20	EA	8A	BE	A7	8C	37	05	19	E7	8C	30	A6	:73
4B60	66	20	86	CC	D3	00	8A	8C	29	20	D4	4F	45	55	49	41	:72
4B70	4D	4B	4E	54	53	48	42	20	44	5A	47	4E	20	59	41	52	:74
4B80	4F	52	45	52	55	52	49	52	41	59	4F	55	59	47	19	54	:84
4B90	80	DE	0A	34	37	30	8D	00	0E	31	06	10	0F	02	EC	49	:1B
4BA0	ED	AA	BD	3D	F0	35	87	0E	00	00	00	00	02	00	00	50	:67
4BB0	A7	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:A7
4BC0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4BD0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4BE0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4BF0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
Sum:	01	E6	E1	4A	94	60	5F	67	65	EB	4E	A0	6B	FE	29	11	:AD
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4C00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4C10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4C20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4C30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4C40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4C50	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4C60	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4C70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4C80	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4C90	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4CA0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4CB0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4CC0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4CD0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4CE0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4CF0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
Sum:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00

## リスト 2

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4D00	16	02	35	16	02	7B	17	02	9B	20	0C	35	10	20	02	8D	:B4
4D10	EF	A6	80	26	FA	6E	84	86	0D	8D	E5	86	0A	20	E1	36	:F3
4D20	06	A6	43	3D	34	06	EC	41	3D	EB	E4	E7	E4	A6	C4	E6	:FA
4D30	43	3D	EB	E0	A6	E0	1E	89	33	44	39	8E	00	00	34	10	:FA
4D40	34	10	34	10	4D	2A	04	8D	40	63	61	6C	60	2B	40	58	:23
4D50	49	2A	F8	44	56	ED	64	EC	C1	2A	04	8D	2C	63	61	6A	:18
4D60	50	2B	1E	58	49	2A	F8	44	56	ED	64	EC	C1	2A	04	8D	:2C
4D70	F8	E3	64	1C	FE	2B	02	1A	01	59	63	69	62	6A	60	2A	:2C
4D80	EC	EC	62	6D	61	32	66	2A	05	43	53	C3	00	01	39	32	:94
4D90	66	CC	7F	FF	39	4D	2B	F1	39	A3	C1	2F	2C	CC	00	00	:16
4DA0	39	A3	C1	2E	24	CC	00	00	39	A3	C1	2D	1C	CC	00	00	:6D
4DB0	39	A3	C1	2C	14	CC	00	00	39	A3	C1	27	0C	CC	00	00	:45
4DC0	39	A3	C1	26	04	CC	00	00	39	C0	00	01	39	83	00	00	:55
4DD0	27	F7	CC	00	00	39	17	FF	2A	1F	89	4F	39	8D	15	16	:48
4DE0	FF	A7	BE	04	3D	17	FF	1E	A6	80	81	24	27	2B	81	2D	:74
4DF0	27	EB	30	1F	CC	00	00	34	06	A6	84	82	30	2B	18	81	:07
Sum:	6D	FD	3F	30	9F	6E	AE	95	2F	B2	5E	EE	0F	01	0C	BF	:31

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4E00	09	22	14	A7	84	EC	60	58	49	58	49	E3	60	58	49	EB	:C7
4E10	80	89	00	ED	60	20	E2	35	B6	CC	00	00	34	06	A6	84	:C2
4E20	82	30	2B	F3	B1	09	23	06	B2	07	B1	0F	22	E9	A7	84	:B6
4E30	EC	60	58	49	58	49	58	49	58	49	58	49	58	49	58	49	:B6
4E40	8E	00	00	36	10	36	10	36	10	36	10	4D	2A	05	63	40	:C5
4E50	17	00	D7	58	49	ED	6E	8E	00	0F	C6	05	1C	FE	A6	C5	:AF
4E60	49	81	09	22	04	1C	FE	20	04	82	00	1A	01	A7	C5	5A	:A4
4E70	26	EC	68	47	69	46	24	02	6C	45	30	1F	26	DC	C6	01	:5F
4E80	A6	C5	26	05	5C	C1	05	26	F7	39	8D	B4	34	04	EB	49	:BB
4E90	C2	06	23	02	8D	20	35	04	6D	40	2A	05	86	2D	17	FE	:77
4EA0	5F	A6	C5	8B	30	17	FE	58	5C	C1	06	26	F4	73	4A	39	:E5
4EB0	33	5E	8D	BC	20	E2	86	20	17	FE	45	5A	26	FA	39	74	:93
4EC0	04	8D	04	25	04	1F	98	1F	89	44	44	44	44	8D	04	1F	:ED
4ED0	98	B4	0F	8B	30	81	39	23	02	8B	07	16	FE	22	1F	98	:44
4EE0	20	F9	33	5C	A6	45	E6	47	3D	ED	42	EC	45	3D	EB	42	:C7
4EF0	89	00	ED	41	A6	44	E6	47	3D	E3	41	ED	41	A6	44	E6	:2D
Sum:	4A	B1	AD	42	3C	E6	90	34	05	57	95	69	AC	1D	21	C4	:A8

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4F00	46	3D	EB	41	89	00	ED	4C	39	74	06	EC	8C	1A	8E	3D	:B9
4F10	09	36	16	8D	BC	33	48	00	00	03	ED	8C	09	35	05	:D5	
4F20	10	36	16	8D	BC	33	48	39	1C	2C	6A	49	16	FE	5A	20	:E3
4F30	7C	FD	34	3E	8C	00	00	00	20	02	20	13	34	31	8D	26	:B3
4F40	10	BE	FC	B2	8E	03	01	AF	A1	A7	A4	8D	2E	35	B1	34	:1E
4F50	05	F6	FD	02	54	25	FA	B7	FD	01	C6	00	F7	FD	00	CA	:A6
4F60	40	F7	FD	00	35	85	34	02	B6	FD	05	2B	FB	1A	40	85	:E2
4F70	80	B7	FD	05	86	FD	05	2A	FB	35	82	7F	FD	05	12	12	:72
4F80	39	34	05	CA	12	8D	59	8D	DD	CC	29	01	FD	FC	82	8D	:98
4F90	EA	BD	D3	BD	02	35	85	86	80	B6	FC	B3	FA	FC	80	F7	:7B
4FA0	FC	80	20	D7	34	35	CC	13	BD	36	8D	BA	CC	04	02	FD	:BE
4FB0	FC	82	CC	11	07	FD	FC	84	BD	C1	8D	AA	F6	FC	83	26	:FF
4FC0	2A	F6	FC	B4	27	12	5A	5A	5A	27	00	10	BE	FC	88	A6	:E3
4FD0	A0	27	05	A7	80	5A	26	F7	C6	0D	E7	84	BD	89	35	85	:D8
4FE0	BD	84	86	0C	FD	FC	B2	20	92	00	00	4F	1F	8B	8D	A7	:FD

## リスト 3

```

1000 CLEAR 300,&H3FFF:DEFINT A-Z:DIM O(60)
1010 WIDTH 80,25:CONSOLE 0,25,0,0:DAT=&H4100
1020 LINE INPUT"HOT OR COLD (H/C)":IN$
1030 IF IN$="h" OR IN$="H" THEN 1050
1040 PRINT "program loading":LOADM"PRM M"
1050 INPUT"フイル / コマ ":NN:DIM F$(NN)
1060 FOR I=1 TO NN:LINE INPUT"FILE NAME ":F$(I):NEXT I
1070 INPUT"GAP ":GP:POKE &H4001,GP
1080 INPUT"FEED ":FE
1084 LINE INPUT"サ ? (y/n)":V$
1085 OPEN"O",#2,"LPT0":
1090 FOR N=1 TO NN
1100 OPEN" I",#1,F$(N):PRINT "OPENED : ";F$(N);
1110 INPUT#1,EM:M=((960-(16+GP)*EM)*32)*2+DAT
1120 IF M<DAT THEN M=DAT
1130 FOR K=DAT TO DAT+120:POKE K,O:NEXT K
1140 WHILE NOT(EOF(1)):L=O:I=M:CR=1:FF=0:J=0
1150 INPUT#1,T:IF V$="Y" THEN T=(T*2)*2*(T+1) MOD 2
1155 POKE &H4000,T
1160 INPUT#1,K

```

## 清書プログラム

```

1170 WHILE K<>-1:L=L+1
1180 POKE I,K*256:POKE I+1,K MOD 256:I=I+2
1190 IF K=0 THEN 1240
1200 IF K=&H222D THEN CR=CR+1:GOTO 1240
1210 IF K=&H222C THEN CR=0:GOTO 1240
1220 IF K=&H222B THEN FF=1:CR=0:GOTO 1240
1230 IF K=&H222A THEN J=L
1240 INPUT#1,K
1250 WEND
1260 POKE &H4002,J+(M-DAT)*#2
1270 EXEC &H5200:PRINT#2,CHR$(27)+A*CHR$(FE);
1271 I=1:WHILE I<=CR:I=I+1:PRINT#2:WEND
1272 FC=FC+FE*2*CR
1273 IF FC>1584 THEN FC=FC-1584
1274 IF FF THEN PRINT#2,CHR$(12);
1280 WEND
1290 CLOSE#1:PRINT" CLOSED"
1300 NEXT
1310 CLOSE:END

```

## リスト 4 清書プログラムダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4D00	16	02	35	16	02	7B	17	02	9B	20	0C	35	10	20	02	8D	:B4
4D10	EF	A6	80	26	FA	6E	84	86	0D	8D	E5	86	0A	20	E1	36	:F3
4D20	06	A6	43	3D	34	06	EC	41	3D	EB	E4	E7	E4	A6	C4	E6	:FA
4D30	43	3D	EB	E0	A6	E0	1E	89	33	44	39	8E	00	00	34	10	:FA
4D40	34	10	34	10	4D	2A	04	8D	40	63	61	6C	60	2B	40	58	:23
4D50	49	2A	F8	44	56	ED	64	EC	C1	2A	04	8D	2C	63	61	6A	:18
4D60	50	2B	1E	58	49	2A	F8	44	56	ED	64	EC	C1	2A	04	8D	:2C
4D70	F8	E3	64	1C	FE	2B	02	1A	01	59	63	69	62	6A	60	2A	:2C
4D80	EC	EC	62	6D	61	32	66	2A	05	43	53	C3	00	01	39	32	:94
4D90	66	CC	7F	FF	39	4D	2B	F1	39	A3	C1	2F	2C	CC	00	00	:16
4DA0	39	A3	C1	2E	24	CC	00	00	39	A3	C1	2D	1C	CC	00	00	:6D
4DB0	39	A3	C1	2C	14	CC	00	00	39	A3	C1	27	0C	CC	00	00	:45
4DC0	39	A3	C1	26	04	CC	00	00	39	C0	00	01	39	83	00	00	:55
4DD0	27	F7	CC	00	00	39	17	FF	2A	1F	89	4F	39	8D	15	16	:48
4DE0	FF	A7	BE	04	3D	17	FF	1E	A6	80	81	24	27	2B	81	2D	:74
4DF0	27	EB	30	1F	CC	00	00	34	06	A6	84	82	30	2B	18	81	:07
Sum:	6D	FD	3F	30	9F	6E	AE	95	2F	B2	5E	EE	0F	01	0C	BF	:31
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4E00	09	22	14	A7	84	EC	60	58	49	58	49	E3	60	58	49	EB	:C7

4E10	80	89	00	ED	60	20	E2	35	B6	CC	00	00	34	06	A6	84	:C2
4E20	82	30	2B	F3	B1	09	23	06	B2	07	B1	0F	22	E9	A7	84	:B6
4E30	EC	60	58	49	58	49	58	49	58	49	58	49	58	49	58	49	:B6
4E40	8E	00	00	36	10	36	10	36	10	36	10	4D	2A	05	63	40	:C5
4E50	17	00	D7	58	49	ED	6E	8E	00	0F	C6	05	1C	FE	A6	C5	:AF
4E60	49	81	09	22	04	1C	FE	20	04	82	00	1A	01	A7	C5	5A	:A4
4E70	26	EC	68	47	69	46	24	02	6C	45	30	1F	26	DC	C6	01	:5F
4E80	A6	C5	26	05	5C	C1	05	26	F7	39	8D	B4	34	04	EB	49	:BB
4E90	C2	06	23	02	8D	20	35	04	6D	40	2A	05	86	2D	17	FE	:77
4EA0	5F	A6	C5	8B	30	17	FE	58	5C	C1	06	26	F4	73	4A	39	:E5
4EB0	33	5E	8D	BC	20	E2	86	20	17	FE	45	5A	26	FA	39	74	:93
4EC0	04	8D	04	25	04	1F	98	1F	89	44	44	44	44	8D	04	1F	:ED
4ED0	98	B4	0F	8B	30	81	39	23	02	8B	07	16	FE	22	1F	98	:44
4EE0	20	F9	33	5C	A6	45	E6	47	3D	ED	42	EC	45	3D	EB	42	:C7
4EF0	89	00	ED	41	A6	44	E6	47	3D	E3	41	ED	41	A6	44	E6	:2D
Sum:	4A	B1	AD	42	3C	E6	90	34	05	57	95	69	AC	1D	21	C4	:A8
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4F00	40	30	21	11	01	89	00	ED	47	38	19	0A	0C	1D	8E	3D	:B9
4F10	09	36	16	8D	BC	ED	42	33	48	1C	00	03	ED	8C	09	35	:D5
4F20	10	36	16	8D	BC	33	48	19	1C	00	49	16	FE	5A	20	1E	:55

4F30	7C	CF	34	3E	8E	00	00	00	20	02	20	12	34	11	9D	26	1E0
4F40	10	BE	FC	8C	8E	03	01	0F	A1	A7	04	8D	2E	75	E1	34	1E1
4F50	05	F6	FD	02	54	25	FA	BC	FD	01	08	00	F7	1D	00	0A	1E5
4F60	40	E7	FD	00	25	85	74	02	86	FD	05	2B	FD	1A	40	8A	1E2
4F70	80	B7	FD	05	85	FD	05	2A	8A	75	82	7F	FD	05	12	12	1E3
4F80	39	34	05	C6	12	8D	52	80	DD	0C	29	01	FD	F0	82	8D	1E8
4F90	EA	8D	D3	8D	02	35	85	C6	80	86	FC	82	F0	1C	80	F7	1E4
4FA0	FC	80	20	D7	34	35	18	1D	3D	78	8D	8A	C0	04	02	FD	1E6
4FB0	FC	B2	CC	11	07	FD	FD	84	80	C1	8B	AA	F0	1C	83	26	1E7
4FC0	2A	F6	FD	34	27	12	5A	5A	5A	27	0D	10	7E	FD	88	AA	1E9
4FD0	A0	27	05	A7	00	5A	26	F7	C6	0D	E7	84	8D	19	25	85	1D0
4FE0	8D	84	86	00	FD	FC	B2	70	20	00	00	4F	1F	8B	8D	A2	1E4
4FF0	86	FD	01	B1	FF	27	06	7D	FD	0F	7E	8F	D0	7E	24	8D	1E5
Sum: DB 0B 8A 12 5E 4C 52 9A 12 1D 12 DC A7 DC E6 84 1E4																	
Add	00	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5100	08	17	F1	6D	39	0A	5E	40	60	84	33	C6	27	CF	04	06	1D0
5110	04	37	44	17	00	84	26	F7	20	DF	30	8D	00	A0	EL	B1	1E1
5120	27	09	10	A3	2E	27	22	A6	80	20	F3	E6	4F	05	E4	70	1D2
5130	54	48	48	48	48	C4	02	27	34	4C	A0	5E	33	C6	73	C8	1D3
5140	04	86	00	34	02	06	C1	C6	00	08	86	84	5A	26	FA	06	1E1
5150	80	5A	F4	26	F0	75	02	20	00	86	84	27	08	60	5F	C5	1E2
5160	01	50	5E	E7	5F	06	4F	30	74	46	16	FF	75	E6	5E	73	1E4
5170	C6	27	C8	D4	96	08	71	08	EL	C1	74	0C	A6	62	75	02	1E7
5180	26	42	58	49	58	49	68	54	59	49	6A	E4	26	F6	00	80	1E5
5190	25	06	86	F8	01	24	DD	16	FF	4F	C6	08	48	24	02	A6	1D7
51A0	F8	04	75	92	2F	14	69	80	5A	26	F1	30	18	06	1B	01	1E9
51B0	4F	04	60	F8	01	24	DD	16	FF	4F	C6	08	48	24	02	A6	1D7
51C0	00	21	40	00	21	40	00	21	40	00	21	40	00	21	40	00	1D1
51D0	21	40	00	21	50	00	21	51	00	21	52	00	21	53	00	21	54
51E0	54	00	21	55	00	21	56	00	21	57	00	21	58	00	21	59	00
51F0	00	21	50	00	21	5B	00	21	5C	01	21	21	01	00	00	00	1D0
Sum: 59 7F 03 25 B2 21 8D CD AA 04 45 70 52 D0 F3 BA 1D1																	
Add	00	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5200	34	71	C6	4C	1F	9B	EC	4C	00	4F	F6	40	00	DD	00	4F	1D4
5210	F6	40	01	DD	02	4F	F6	40	02	DD	04	CC	41	00	36	06	1D7
5220	DC	02	36	06	CC	01	00	BD	40	1F	D3	04	36	06	DC	00	1E1
5230	26	06	CC	01	00	BD	40	1F	34	06	34	10	34	10	34	10	1E3
5240	34	10	BD	50	00	12	58	75	FF	C8	00	1F	9B	7E	41	F0	1E5
5250	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	1D2
Sum: 70 D7 86 B0 ED DA 77 9D F0 B6 FF 39 46 71 9B 55 1E4																	

## リスト 5 Kソース・リスト(参考)

```

1000 * (*FM-B JAPANESE WORD PROCESSOR BY T.SHIGEMURA*)
1010 DEF RR=$4AD7;DEF CODE=$48B3;DEF LPRINT=$46B4
1020 DEF KK=$3E00;DEF WW=$5000;DEF SC=$505A;DEF LL=$4B93
1030 CONST YM=9;RM=$07;KM=$1B;FM=38;EL=39
1040 CONST LF=$222D;BR=$222C;SD=$222A;FF=$222B
1050 BADR=PEEK($4BAF)*256+PEEK($4BB0);OUTBF=BADR+796
1060 BU1=BADR+916;SN=PEEK(BADR+4);TEXT=BADR+SN+926
1070 FNC=BADR+6;FNK=BADR+16;EM=PEEK(BADR+1)
1080 SCL=EM-EL-1;SCR=EL-SCL-1;X=0;Y=0;SX=0;SY=0;TB=8;KC=0;CSW=-1
1090 KNM=0;KN=0;RJ=0;KT=0;INM=0;IND=0;XM=EL-1;ADR=SCL+1;CL=7
1100 FOR I=0 TO 9;FNC(I)=0;NEXT
1110 J=PEEK(BADR+2)*256+PEEK(BADR+3)-TEXT
1120 FOR I=J/2+1 TO POINT(XM,YM,1,SN)
1130 TEXT(I)=0
1140 NEXT;CLS[]
1150 REPEAT;CURS[];IN=GET;CURS[]
1160 IF IN=$7F THEN DEL[];FI
1170 IF IN=$08 THEN X=X-1;CURS[];DEL[];FI
1180 IF IN=$1C THEN X=X+1;CURS[];FI
1190 IF IN=$1D THEN X=X-1;CURS[];FI
1200 IF IN=$1E THEN Y=Y-1;CURS[];FI
1210 IF IN=$1F THEN Y=Y+1;CURS[];FI
1220 IF IN=$02 THEN X=X-5;CURS[];FI
1230 IF IN=$06 THEN X=X+5;CURS[];FI
1240 IF IN=$19 THEN IF X=0 THEN Y=Y-3 ELSE X=0;Y=Y-1;FI;CURS[];FI
1250 IF IN=$1A THEN IF X=0 THEN Y=Y+3 ELSE X=0;Y=Y+1;FI;CURS[];FI
1260 IF IN=$16 THEN IF X=0;N=POINT(X,Y,0,SY);P=POINT(XM,Y,1,SY)
1270 FOR J=N TO P;U=TEXT(J)
1280 IF U<>0 AND U<>LF AND U<>BR THEN E=J
1290 IF I=-1 THEN I=J;FI
1300 FI
1310 NEXT
1320 IF I<>-1 THEN
1330 N=(N+P)/2;O=(I+E)/2;INM=-1;X=0;CHCL[]
1340 IF O>N THEN FOR P=N TO O-1;DEL[];NEXT
1350 ELSE FOR P=O TO N-1;DISP[O];X=0;NEXT
1360 FI
1370 INM=IND
1380 FI
1390 FI
1400 IF IN=$14 THEN;CSW=0
1410 IF Y+SY<YM+SN THEN INM=-1;P=X;IF SX<>0 THEN P=P+ADR;FI
1420 N=-1;O=-1;E=POINT(X,Y+1,0,SY);J=E
1430 FOR I=POINT(X,Y,SX,SY) TO POINT(XM,Y,1,SY)
1440 U=TEXT(I)
1450 IF U=0 OR U=LF OR U=BR THEN
1460 IF U=LF OR U=BR THEN TEXT(I)=0;FI
1470 IF N=-1 THEN N=I;FI
1480 ELSE N=-1
1490 IF E=J THEN E=I;FI
1500 FI
1510 NEXT
1520 IF N<>-1 THEN E=N-E
1530 N=J-N-1
1540 ELSE E=J-E;FI

```

```

1550 *      FOR O=0 TO N
1560 *      DISP[O]
1570 *      NEXT;CHCL[];X=0;Y=Y+1;CUR[]
1580 *      N=POINT[XM,Y,1,SY];O=0
1590 *      WHILE O<E AND TEXT(N)=0
1600 *          DISP[TEXT(N-EM)]
1610 *          FOR K=N-EM TO N-EM-EM+P+2 STEP -1
1620 *              TEXT(K)=TEXT(K-1);TEXT(K-1)=0
1630 *          NEXT
1640 *          FOR K=0 TO XM
1650 *              WRITE[K,Y-1,TEXT(N-EM-EM+K+1)]
1660 *          NEXT
1670 *          O=O+1
1680 *      WEND;INM=IND
1690 *      IF O<E THEN PRINT CHR$(7)
1700 *          X=XM-E+O+1
1710 *          IF X>=0 THEN SX=-1;SC2[]
1720 *          ELSE X=X+ADR
1730 *      FI
1740 *      ELSE TEXT(N-EM)=0
1750 *      FI
1760 *      FI;CSW=-1
1770 *  FI
1780 *  IF IN=$17 THEN
1790 *      PRINT"TYPE = ";N=INPUT
1800 *      IF N>3 THEN O=0;X=0;CHCL[]
1810 *      E=POINT[O,Y,O,SY];INM=-1;P=0
1820 *      WHILE (X<XM OR SX=0) AND X<EM-1;L=TEXT(E+P)
1830 *          IF NOT(L=0 OR L=LF OR L=BR OR L=SO OR TEXT(E+EM-1)<>0) THEN
1840 *              IF TEXT(E+P-1)<>SO OR P=0 THEN
1850 *                  IF X>XM THEN SX=1;CLSC[];X=X-EL+SCR;FI
1860 *                  DISP[SO];P=P+1;O=X
1870 *              FI
1880 *              FI;X=X+1;P=P+1
1890 *          WEND;X=0;INM=IND
1900 *          ELSE IF BU1:Y+SY)<>0 THEN SB[Y];FI
1910 *          BU1:Y+SY)=N
1920 *          IF BU1:Y+SY)<>0 THEN SB[Y];FI
1930 *      FI
1940 *  FI
1950 *  IF IN=$18 THEN PRINT"FUNCTION KEY NUMBER = ";IN=INPUT
1960 *      IF IN<11 AND IN>0 THEN E=IN-1;O=0
1970 *      REPEAT;CUS2[]
1980 *          IF IN=$11 THEN
1990 *              FNK(FM*E+O)=TEXT(POINT[X-1,Y,SX,SY])
2000 *              O=O+1;FNC:E)=O;FI
2010 *          UNTIL IN<>$11 OR O>=FM
2020 *          IF IN=$11 THEN
2030 *              REPEAT PRINT CHR$(7)
2040 *              CURS[];IN=GET;CURS[]
2050 *              UNTIL IN<>$11
2060 *              ELSE X=X-1
2070 *          FI
2080 *          ELSE ERASE[]
2090 *          FOR I=1 TO 10;K=I-(I/10)*10;L=I-1
2100 *              SYMBOL[O,I*18-18,K+$30,4]
2110 *              FOR J=1 TO FNC:L)
2120 *                  WRITE[J,I-1,FNK(FM*L+J-1)]
2130 *              NEXT
2140 *          NEXT;PRINT"OK ?";IN=GET;CLSC[]
2150 *      FI;IN=O
2160 *  FI
2170 *  IF IN=$15 THEN O=GET
2180 *      IF O<11 THEN O=O-1
2190 *      FOR N=1 TO FNC:O)
2200 *          DISP[FNK(FM*O+N-1)]
2210 *      NEXT
2220 *  FI
2230 *  FI
2240 *  IF IN=$0C THEN PRINT "CLEAR = 1  CONTINUE = 9  (1/9)? ";I=INPUT
2250 *      IF I=1 THEN
2260 *          FOR J=0 TO POINT[XM,YM,1,SN];TEXT(J)=0;NEXT
2270 *          FOR J=0 TO SN+YM;BU1:J)=0;NEXT
2280 *          SX=0;SY=0;X=0;Y=0;CLSC[]
2290 *      FI
2300 *  FI
2310 *  IF IN=$11 THEN
2320 *      IF Y>0 OR SY>0 THEN INM=0;I=POINT[X,Y-1,SX,SY]
2330 *      DISP[TEXT(I)]
2340 *      INM=IND
2350 *  FI
2360 *  FI
2370 *  IF IN=$0D THEN
2380 *      X=0;Y=Y+1;CUR[];CHCL[]
2390 *  FI
2400 *  IF IN=$0B THEN

```

```

2410 *      IF X=0 AND Y=0 THEN SY=0;SX=0;CLS[]
2420 *      ELSE X=0;Y=0
2430 *      FI
2440 *  FI
2450 *  IF IN=$0E THEN S3[];KT=CHANGE[KT];FI
2460 *  IF IN=$12 THEN S4[];INM=CHANGE[INM];IND=INM;FI
2470 *  IF IN=$10 THEN
2480 *    PRINT"PRINT=0   77"セット=2   コート=4   カンシ"モート=8   "
2490 *    IN=INPUT
2500 *    IF IN=8 THEN PRINT"TYPE 1 = 1   TYPE 2 = 2 ";I=INPUT
2510 *      IF I=2 THEN KNM=-1
2520 *      ELSE KNM=-0
2530 *    FI
2540 *  FI
2550 *  IF IN=4 THEN PRINT"CODE = ";I=INPUT;DISP[I];FI
2560 *  IF IN=2 THEN PRINT"TAB = ";I=INPUT
2570 *    IF I<2 THEN I=2;FI
2580 *    IF I>50 THEN I=50;FI
2590 *    TB=I
2600 *  FI
2610 *  IF IN=0 THEN PRINT"GAP = ";GP=INPUT
2620 *    IF GP>60 OR GP<0 THEN GP=4;FI
2630 *    PRINT"FEED = ";FEC=INPUT
2640 *    IF FEC>60 OR FEC<1 THEN FEC=12;FI
2650 *    PRINT"77 = 1   33 = 9 ";VE=INPUT
2660 *    IF VE<>1 THEN VE=0;FI
2670 *    E=0;J=0;K=0;O=0
2680 *    FOR I=0 TO SN+YM;L=POINT[O,I,0,0]
2690 *      FOR M=0 TO EM-1;U=TEXT(L+M)
2700 *        IF U<>0 THEN E=I;FI
2710 *        IF U=50 THEN
2720 *          IF J THEN K=-1
2730 *          ELSE J=-1
2740 *        FI
2750 *        ELSE J=0
2760 *      FI
2770 *    NEXT
2780 *    NEXT;M=(960-(16+GP)*EM)/32;IF M<1 THEN M=0;FI
2790 *    FOR I=0 TO EM-1
2800 *      OUTBF(I)=0
2810 *    NEXT
2820 *    IF K THEN PRINT CHR$(7),"ERR DATA";IN=GET;PRINT/
2830 *    ELSE FOR I=0 TO E;N=0;K=1;S=0;L=POINT[O,I,0,0]
2840 *      FOR J=0 TO EM-1;U=TEXT(L+J)
2850 *        ,IF U<>0 THEN
2860 *          IF U<>LF THEN
2870 *            IF U=BR THEN K=0
2880 *            ELSE
2890 *              IF U=FF THEN
2900 *                S=-1
2910 *              ELSE N=J+1
2920 *            FI
2930 *            IF
2940 *              ELSE K=K+1
2950 *            FI
2960 *          FI
2970 *        NEXT;R=BU1:I)
2980 *        IF VE=1 THEN
2990 *          IF R-(R/2)*2=0 THEN R=R+1
3000 *          ELSE R=R-1
3010 *        FI
3020 *      IF TEXT(L+N-1)=50 AND N>0 THEN N=N-1;FI
3030 *      IF S THEN LL[$0C]
3040 *      ELSE FOR O=0 TO N-1
3050 *        OUTBF(M+O)=TEXT(L+O)
3060 *      NEXT
3070 *      LPRINT[OUTBF,GP*256+N+M,R*256]
3080 *      FOR J=1 TO K;FEED[FEC];NEXT
3090 *    FI
3100 *  NEXT
3110 *  FI
3120 *  FI;IN=0
3130 *  FI
3140 *  I=POINT[X,Y,SX,SY]
3150 *  IF (IN>$20 AND IN<$41) OR (IN>$5A AND IN<$61) THEN
3160 *    DISPCODE[IN*256]]
3170 *  FI
3180 *  IF (IN>$7A AND IN<$7F) OR (KN=0 AND IN=$20) THEN
3190 *    DISPCODE[IN*256]]
3200 *  FI
3210 *  IF (IN>$7F AND IN<$A6) OR IN=$B0 OR (IN>$DD AND IN<$FF) THEN
3220 *    DISPCODE[IN*256]]
3230 *  FI
3240 *  IF IN>$A6 AND IN<$B0 THEN
3250 *    DISPCODE[IN*256-KT]]
3260 *

```

```

3270  FI
3280  IF IN=$09 THEN INM=0;N=(X/TB)*TB+TB
3290      IF N>XM THEN N=XM;FI
3300      FOR O=X TO N;DISP[O];NEXT
3310      INM=IND;FI
3320  IF IN=$0A THEN J=POINT[XM,YM,1,SN];I=0
3330      FOR K=J-EM+1 TO J;IF TEXT(K)<>0 THEN I=J+1;FI;NEXT
3340      IF I+EM>J THEN PRINT CHR$(7)
3350          ELSE M=POINT[O,Y+1,0,SY];L=M
3360          FOR K=J TO M+EM STEP -1
3370              TEXT(K)=TEXT(K-EM)
3380          NEXT
3390          FOR K=M TO M+EM-1;TEXT(K)=0;NEXT
3400          I=POINT[X,Y,SX,SY]
3410          FOR K=I TO M-1
3420              TEXT(L)=TEXT(K);TEXT(K)=0;L=L+1
3430          NEXT
3440          WRT2[ ];Y=Y+1;X=0;CUR[ ];CHCL[ ]
3450      FI
3460  FI
3470  IF IN=$0F THEN
3480      IF SY>0 OR Y>0 THEN Q=POINT[O,Y,0,SY]
3490          IF I<>Q THEN FOR P=1 TO X-SX*EL
3500              X=X-1;CUR[ ];DEL[ ]
3510          NEXT
3520          FI
3530          J=POINT[XM,YM,1,SN]
3540          FOR K=Q TO J;TEXT(K-EM)=TEXT(K);NEXT
3550          FOR K=J-EM+1 TO J;TEXT(K)=0;NEXT
3560          Y=Y-1;CUR[ ];WRT2[ ];CHCL[ ]
3570          ELSE PRINT CHR$(7)
3580      FI
3590  FI
3600  IF IN=$05 THEN J=POINT[XM,Y,1,SY]
3610      FOR L=I TO J;TEXT(L)=0;NEXT
3620      IF IN=8 THEN PRINT"TYPE1 = 1 TYPE2 = 2 ";IN=INPUT
3630          IF IN=1 THEN KNM=0;FI
3640          IF IN=2 THEN KNM=-1;FI
3650      FI
3660      FOR L=X TO XM;WRITE[L,Y,0];NEXT
3670  FI
3680  ALP[IN]
3690  IF RJ THEN IF BF OR SF THEN ROMA[ ];FI
3700      ELSE IF BF OR SF THEN DISPCODE[IN*256-2*GR];FI;FI
3710  IF (IN>$B0 AND IN<$DE) OR IN=$A6 THEN KF=-1 ELSE KF=0;FI
3720  IF KN THEN
3730      IF KF OR (IN=$20 AND KC) THEN KANJI[ ]
3740      ELSE;IF IN=$20 THEN DISP[0];FI
3750  FI
3760  ELSE IF KF THEN DISPCODE[IN*256-KT];FI
3770  FI
3780  INM=0
3790  IF IN=$04 THEN DISP[TEXT(I)-1];FI
3800  IF IN=$01 THEN DISP[TEXT(I)+1];FI;INM=IND
3810  IF IN=$1B THEN S5[ ];KN=CHANGE[KN];LINE[59,182,67,190,1,7,2];KC=0;FI
3820  IF IN=$RM THEN S2[ ];RJ=CHANGE[RJ];FI
3830  IF IN=$03 THEN
3840      PRINT"Sure? (1) ";IN=INPUT
3850      IF IN=1 THEN IN=3 ELSE IN=0;FI
3860  FI
3870  'UNTIL IN=3
3880  E=0
3890  FOR I=0 TO POINT[XM,YM,1,SN]
3900      IF TEXT(I)<>0 THEN E=I*2;FI
3910  NEXT
3920  POKE BADR+2,HIGH(E+TEXT);POKE BADR+3,LOW(E+TEXT)
3930  CODE $35,$FF
3940  'FEED; LL[$1B*256+$41];LL[%1*256+$0D];LL[$0A];RETURN
3950  'CUS2;CUR[ ];CUD[ ];IN=GET;CUD[ ];X=X+1;RETURN
3960  'CHCL;IF SX THEN SX=0;SC2[ ];FI;RETURN
3970  'CHANGE;IF %1=0 THEN R=-1 ELSE R=0;FI;RETURN
3980  'KANJI;IF KC=0 THEN KC=1;KB=IN;S6[ ]
3990      ELSE KC=0;S6[ ];K=KK[KB*256+IN];IN=0
4000      IF K<>0 THEN L=PEEK($3E20);P=L;J=K
4010      IF L<>1 THEN
4020          IF KNM<>0 THEN
4030              WHILE IN<>$0D AND IN<>$1F;WRITE[X,Y,J];IN=GET
4040              IF IN=$1D OR IN=$0B THEN J=J-1;L=L+1;FI
4050              IF IN=$20 THEN J=J+1;L=L-1;FI
4060              IF J<K THEN J=K;L=L-1;PRINT CHR$(7);FI
4070              IF L<=0 THEN J=J-1;L=1;PRINT CHR$(7);FI
4080              J=KCH[ ]
4090          WEND
4100          ELSE N=-1;O=K;IN=$1F
4110          LINE[0,180,639,199,1,7,2]
4120          WHILE IN<>$0D AND IN<>$20

```

```

4130 * IF IN=$1F THEN
4140 * IF L>0 THEN N=N+1;J=0;R=0
4150 * FOR I=0 TO XM
4160 * KNSUB[0]
4170 * L=L-1;O=KCCH[0+1]
4180 * NEXT
4190 * ELSE PRINT CHR$(7)
4200 * FI
4210 * FI
4220 * IF IN=$1E THEN
4230 * IF N>0 THEN N=N-1;R=0;L=L+EL;O=J
4240 * FOR I=XM TO 0 STEP -1
4250 * J=KCCH[J-1]
4260 * KNSUB[J]
4270 * NEXT
4280 * ELSE PRINT CHR$(7)
4290 * FI
4300 * FI
4310 * IF IN=$1C THEN R=R+1
4320 * IF R>XM THEN R=0;FI
4330 * FI
4340 * IF IN=$1D THEN R=R-1
4350 * IF R<0 THEN R=XM;FI
4360 * FI
4370 * CUDS[R,YM+1];IN=GET;CUDS[R,YM+1]
4380 * WEND
4390 * FOR I=1 TO R;J=KCCH[J+1];NEXT
4400 * IF N*EL+R>P THEN IN=0;FI
4410 * FOR I=0 TO XM;WRITE[I,YM+1,0];NEXT
4420 * BOX[]
4430 * FI
4440 * ELSE IN=$0D
4450 * FI
4460 * M=TEXT(POINT[X,Y,SX,SY])
4470 * IF IN=$0D THEN DISP[J] ELSE WRITE[X,Y,M];FI
4480 * FI
4490 * FI;IN=0;RETURN
4500 * KNSUB;IF L>0 THEN M=%1
4510 * ELSE M=0
4520 * FI
4530 * WRITE[I,YM+1,M]
4540 * RETURN
4550 * KCCH; Q=%1;M=LOW(%1)
4560 * IF M=$7F THEN Q=%1+$A2;FI
4570 * IF M=$20 THEN Q=%1-$A2;FI
4580 * Q=Q;RETURN
4590 * ALP; IF %1>$40 AND %1<$5B THEN BF=-1 ELSE BF=0;FI
4600 * IF %1>$60 AND %1<$7B THEN SF=-1 ELSE SF=0;FI
4610 * RETURN
4620 * SC2; IF SX=0 THEN I=$8080
4630 * ELSE I=$8F8F
4640 * FI
4650 * SC[I,EM];T=POINT[0,0,SX,SY]
4660 * IF SX=0 THEN FOR I=SCL TO 0 STEP -1
4670 * SC3[]
4680 * NEXT
4690 * ELSE L=L+SCR
4700 * FOR I=SCR TO XM
4710 * SC3[]
4720 * NEXT
4730 * FI;S9[];RETURN
4740 * SC3; L=T+I
4750 * FOR J=0 TO YM
4760 * IF TEXT(L)<>0 THEN WRITE[I,J,TEXT(L)];FI
4770 * L=L+EM
4780 * NEXT
4790 * RETURN
4800 * SRO; SC[%1]
4810 * IF %1 THEN L=0 ELSE L=YM;FI
4820 * J=POINT[0,L,SX,SY];IF BUI:L+SY)<>0 THEN SB[L];FI
4830 * FOR K=0 TO XM;R=TEXT(J+K);IF R THEN WRITECK[L,R];FI;NEXT
4840 * RETURN
4850 * CLS; ERASE[];BOX[]
4860 * FOR I=0 TO YM;L=POINT[0,I,SX,SY];IF BUI:I+SY)<>0 THEN SB[I];FI
4870 * FOR J=0 TO XM;WRITE[J,I,TEXT(L+J)];NEXT
4880 * NEXT;RETURN
4890 * POINT; R=%1+(%2+%4)*EM;IF %3 THEN R=R+ADR;FI;R=R;RETURN
4900 * WRITE; WW[%3,%1+1,%2];RETURN
4910 * CHE; R=0
4920 * IF %1='A' OR %1='I' OR %1='U' OR %1='E' OR %1='O' THEN R=-1;FI
4930 * IF %1='a' OR %1='i' OR %1='u' OR %1='e' OR %1='o' THEN R=-1;FI
4940 * R=R;RETURN
4950 * ROMA; IF CHE[IN] THEN I=$20;RHE[]
4960 * ELSE I=IN;L=0
4970 * REPEAT IN=I;S7[IN];I=GET;S7[IN]
4980 * IF (CHE[IN] OR IN='N' OR IN='n') AND I=' ' THEN L=-1;FI

```

```

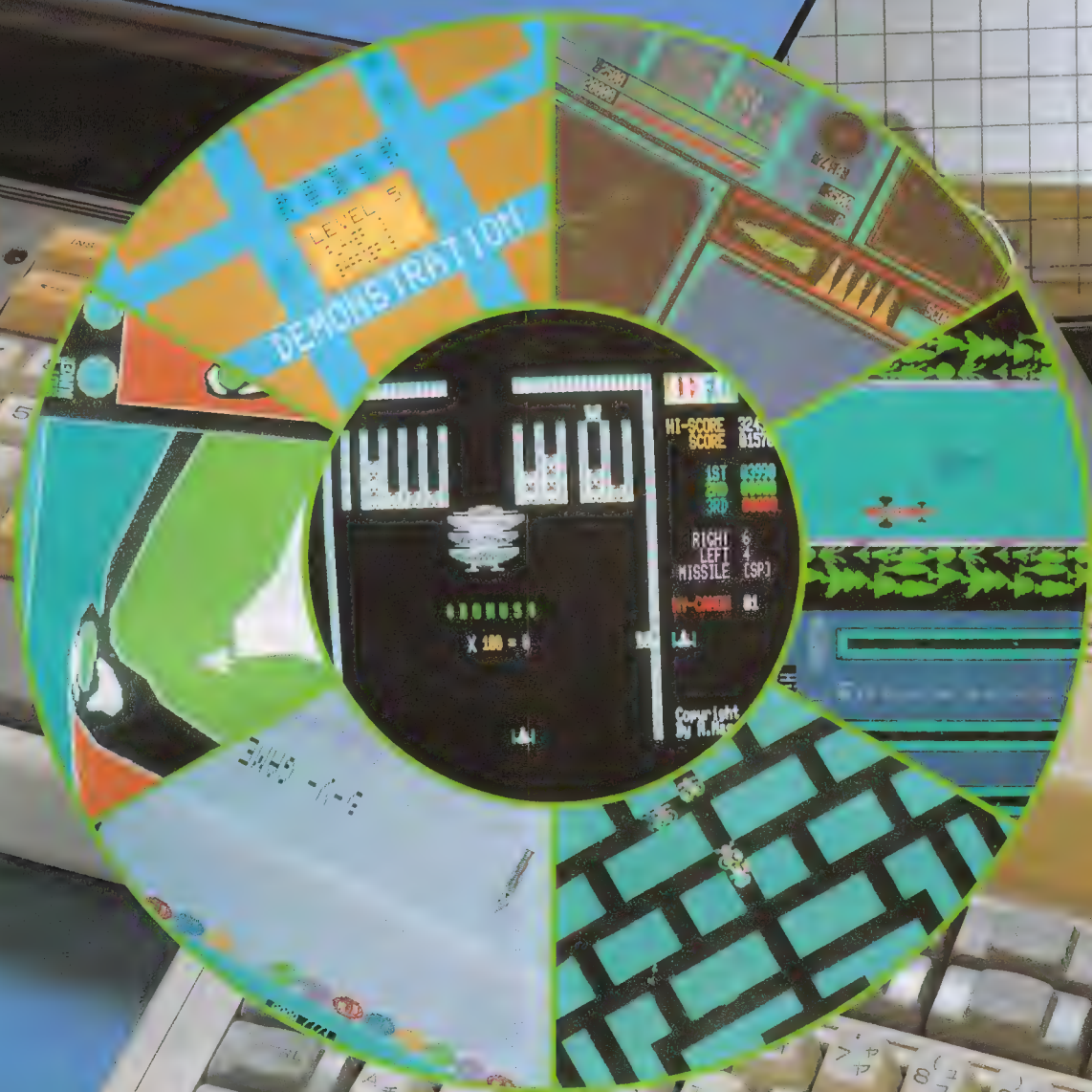
4990 UNTIL CHECKJ OR I=RM OR L OR I=KM
5000 IF IN= THEN L=IN; IN=I; I=L; FI
5010 IF I=KM THEN IN=KM
5020 ELSE IF I=RM THEN IN=RM
5030 ELSE RHEI
5040 FI
5050 FI
5060 FI; RETURN
5070 RHE; ALP[IN]; IF SF THEN IN=IN-$20; FI
5080 ALP[I]; IF SF THEN I=I-$20; FI
5090 IN=RR[IN*256+I]; I=LOW(IN); IN=HIGH(IN)
5100 IF KN=0 THEN IN=CODE[IN*256-KT]
5110 IF I=$DE THEN IN=IN+1; FI
5120 IF I=$DF THEN IN=IN+2; FI
5130 DISP[IN]; IN=0
5140 FI; RETURN
5150 DISP; I=POINT[X,Y,SX,SY]
5160 IF INM THEN J=POINT[XM,Y,1,SY]
5170 IF TEXT(J)<>0 THEN PRINT CHR$(7)
5180 ELSE IF J<>I THEN
5190 FOR K=J-1 TO I STEP -1
5200 TEXT(K+1)=TEXT(K)
5210 NEXT
5220 FOR K=XM TO X+1 STEP -1
5230 WRITE[K,Y,TEXT(I+K-X)]
5240 NEXT
5250 FI
5260 FI
5270 FI
5280 TEXT(I)=%I; WRITE[X,Y,%I]
5290 IF CSW=-1 THEN X=X+1; FI
5300 IF X>XM AND SX THEN Y=Y+1; FI
5310 CUR[ ]; RETURN
5320 WRT2; FOR K=Y TO YM; I=POINT[0,K,SX,SY]
5330 FOR J=0 TO XM; WRITE[J,K,TEXT(I+J)]; NEXT
5340 NEXT; RETURN
5350 CUR; IF X>XM THEN IF SX THEN X=0 ELSE X=SCR; FI
5360 SX=CHANGE[SX]; SC2[ ]; FI
5370 IF X<0 THEN IF SX THEN X=SCL ELSE X=XM; FI
5380 SX=CHANGE[SX]; SC2[ ]; FI; M=0
5390 IF Y>YM THEN FOR I=1 TO Y-YM; SY=SY+1
5400 IF SY<=SN THEN M=-1; SROI[0] ELSE SY=SN; FI
5410 NEXT
5420 IF M THEN BOX[ ]; FI
5430 Y=YM; FI
5440 IF Y<0 THEN FOR I=-1 TO Y STEP -1; SY=SY-1
5450 IF SY>=0 THEN M=-1; SROI[-1] ELSE SY=0; FI
5460 NEXT
5470 IF M THEN BOX[ ]; FI
5480 Y=0; FI
5490 RETURN
5500 DEL; I=POINT[X,Y,SX,SY]; J=POINT[XM,Y,1,SY]
5510 FOR K=I TO J-1; TEXT(K)=TEXT(K+1); NEXT; TEXT(J)=0
5520 FOR K=0 TO XM-X; WRITE[X+K,Y,TEXT(I+K)]; NEXT; RETURN
5530 S2; SYMBOL[600,182,'*',4]; RETURN
5540 S3; SYMBOL[610,182,'*',4]; RETURN
5550 S4; SYMBOL[630,182,'*',4]; RETURN
5560 S5; SYMBOL[620,182,'*',4]; RETURN
5570 S6; SYMBOL[59,182,KB,4]; RETURN
5580 S7; SYMBOL[82,182,%I,4]; RETURN
5590 S8; CL=4; SYMBOL[0,%I*18,BUI:SY+%I)+$30,4]; CL=7; RETURN
5600 S9; SYMBOL[52,182,'*',4]; RETURN
5610 BOX; LINE[50,181,91,191,0,7,1]
5620 LINE[599,181,639,191,0,7,1]
5630 LINE[609,181,609,191,0,7,0]
5640 LINE[619,181,619,191,0,7,0]
5650 LINE[629,181,629,191,0,7,0]
5660 IF SX THEN S9[ ]; FI
5670 IF RJ THEN S2[ ]; FI
5680 IF KT THEN S3[ ]; FI
5690 IF IND THEN S4[ ]; FI
5700 IF KN THEN S5[ ]; FI
5710 IF KC THEN S6[ ]; FI
5720 RETURN
5730 CUD; LINE[X*16+16,Y*18+15,X*16+31,Y*18+17,4,7,1]; RETURN
5740 CURS; CUDS[X,Y]; RETURN
5750 CUDS; LINE[X*16+16,X*18+16,X*16+31,X*18+16,4,7,0]; RETURN
5760 ERASE; HLT[ ]; ADD=$FCB2; CPOKE ADD,13,0,0,7; SUB[ ]; RETURN
5770 SYMBOL; HLT[ ]; ADD=$FCB2; CPOKE ADD,25,CL,%4,0,1,1,X1#,X2#,1,X3; SUB[ ]; RETURN
5780 LINE; HLT[ ]; ADD=$FCB2; CPOKE ADD,21,%6,%5,X1#,X2#,X3#,X4#,X5; SUB[ ]; RETURN
5790 HLT; CODE$B6,$FD,$05,$2B,$FB; POKE $FD05,$B0
5800 CODE$B6,$FD,$05,$2A,$FB; RETURN
5810 SUB; POKE $FD05,0; RETURN

```



## ゲーム編

FM-7/8 ALIEN FALL .....	正岡伸博	162
FM-7/8 キンチョマン .....	服部浩憲	170
FM-7 3-Dゴルフ・シミュレーション .....	T&Eソフト	175
FM-8 戦車最前線 .....	高畑英樹	185
FM-7/8 3次元迷路 .....	市川浩志	189
FM-8 FMラリー .....	高畑英樹	193
FM-8 日本復活 .....	マイクロテクノロジー研究所	196
FM-8 シーソーゲーム .....	高畑英樹	203

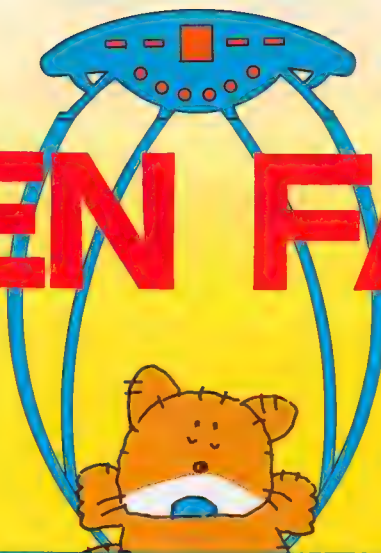




FM-7/8

# ALIEN FALL

エイリアン・フォール



■正岡伸博

このゲームの原作は1981年11月号、芸夢狂人氏作成の『ALIEN FALL』です。FMシリーズの機能をフルに活かして、フルカラー、PSGサウンド、サブCPUとのパレル動作による高速処理という特徴があります。

## ● プログラムについてその1

このプログラムは大きく次の4つに分かれています。

2D00H~34FFH：PSGコントロール、説明、デモ  
3500H~3FFFH：本ゲーム専用ゲームBIOS  
4000H~4FFFH：サブCPUコントロール（サブCPUに転送する）  
5000H~58FFH：ゲーム・メイン・プログラム

この方式の特徴を次に掲げます。

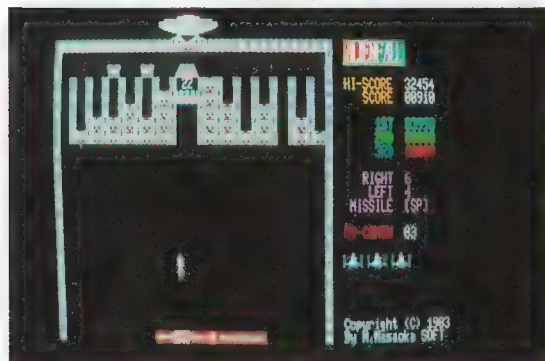
### 長所

- ①各ルーチンがコンパクトになり、働きが明確になる。
- ②分割してアSEMBルしやすくなる。
- ③汎用性がある（特にサブCPUコントロールなどは少しの改造で他のゲームに流用できる）。

### 短所

- ①プログラムが大きくなる（このプログラムで10Kバイト以上）。
- ②各ルーチンを結び付ける方法に工夫がいる（スピードを採るか、デバッグの容易さを採るか）。

やられた！



以上簡単に挙げましたが、両者を天秤にかけると、長所の方が勝るのでは？

## ● プログラムについてその2

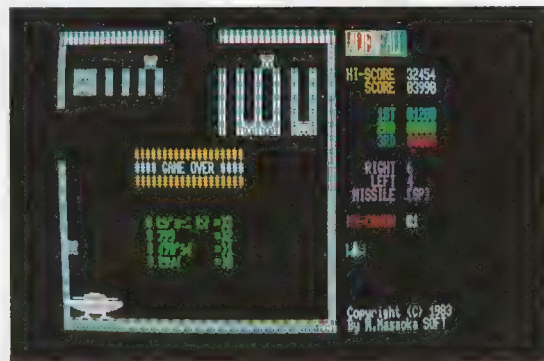
このプログラムは外観上、原作とほぼ同じだと思います（原作をやったことがないので詳細は不明）。しかしアルゴリズムでは異なります。たとえばFALL ALIEN（以下FAと略す）ですが、原作ではI、II、IIIの3分割でしたが、ここではI（中央の穴から1行下がった所まで）、II（左右への横移動）、III（溝へ落ち、待機する）、IV（下へ降りる）の4つに分割しています。また、画面中央の数字は原作では局面を示していましたが、ここでは撃ち落としたFAの数になっています。

スピードについては相当ディレイをかけているため、初めは簡単ですが、面が進むにつれて高速になり、難しくなるので安心してください。最初オール・カラーにしたことによりスピードが落ちるのではという心配があったのですが、実際作成してみるとさすがFMシリーズのマルチCPUは強く、FMZのようにキャラクタを使ったゲームにも匹敵するほどのスピードになり、安心しました。

## ● 遊び方

ビーム砲はテン・キーの[4]で左へ、[6]で右へ移動します。ビームの発射は[SPACE]キーです。

あえなくGAME OVER





右側からBASE UFOが出てきて中央に来るとFAを降ろし始めます。FAは溝に入り、ここへ6匹めが入ろうとすると1番下のFAは押し出されて降下してきます。また一番端へ行ったFAはすぐに降下してきます。

1番下へ落ちるとFAは衝撃波を出すので、その前に破壊するか逃げるかしてください。ときどき中央からUFOが降りてきます。これは一番下へ来ると全面に衝撃波を出すためやっつけないと必ずやられてしまいます。

50匹のFAがやられるとBASE UFOはデモンストレーションの後、左へ移動して1面の終了です。

## 得点

溝に待機中のFAは20点、降下しているFAは40点、UFOはミステリーで100~300点です。

得点は約6万点までしかありません。一番下の桁をタミーにして約60万点までにするのは非常に簡単なのですが、今回あえてしませんでした。ゲームもホドホドにということだと思ってください。

表1 仮想VRAM内での表示(下記以外は座標で判断する) 表2 PSGルーチン・コール時のAccA

内 容	表 示
空 白	0
FALL ALIEN	3, 83
MY CANNON	5
MY CANNON(爆発)	6
壁	8
UFOからの衝撃波	10~1 F
FALL ALIEN	20~2 F

AccA	内 容
1	MY CANNON爆発
2	MISSILE発射
3	FALL ALIEN-MISSILE爆発
4	UFOから衝撃波発生
5	UFO-MISSILE爆発
6	FALL ALIENから衝撃波発生
7	BASE UFO移動
8	UFO発生
9	ボーナス
A	ゲーム・オーバー

表3 サブルーチンの内容(\$5000~)

アドレス	内 容
504E	衝撃波消去
5060	FALL ALIEN発生
5072	FALL ALIEN移動
519A	MY CANNON移動, MISSILE発射
5229	MISSILE上昇
5279	MYSTERY UFO発生, 移動
531A	ディレイ
5328	FALL ALIEN生死判定
5331	UFO生死判定
5347	MISSILE生死判定
5361	MY CANNON生死判定
536C	MY CANNONが衝撃波にやられたとき
5389	FALL ALIENとMISSILEが爆発
53C4	UFOとMISSILEが爆発
53E3	FALL ALIENのスタックをクリア
53F3	FALL ALIENが下に落ちたとき
5406	UFOが下に落ちたとき
5425	次の面に進む
54CF	MY CANNONがやられた
55A6	ゲーム・オーバー
5697	BASE UFOの色変化
56A3	タイトルの点滅
56C6	次の面へ進むときの初期設定
5704	MY CANNONがやられたときの初期設定
571A	画面を描く
573C	ゲームが始まったときの初期設定
5784	POWER UP

表5 サブ側と通信する際のコマンドの詳細

コールNo.	ラベル	FC82	FC83	FC84	FC85	FC86	備 考
1	CLS0						普通のCLS
2	CLS1						上下カーテン式CLS
3	CHGCOL						B→R→G変化
4	INKEY	キーデータ					
5	PRINT	X	Y	ASCII		カラー	
6	DECPR5	"	"	HEX(H)	HEX(L)	"	5桁出力
7	DECPR2	"	"		HEX	"	2桁出力
8	CHER	"	"				1×1
9	SHER	"	"				1×1
A	MMPR	"	"				1×1
B	FAPR	"	"				2×1
C	FAER	"	"				2×1
D	UFPR	"	"				4×1
E	UFER	"	"				4×1
F	BAPR	"	"				10×2
10	BAER	"	"				10×2
11	MEPR	"	"				3×1
12	MCER	"	"				3×1
13	MCBPR	"	"				3×1
14	UFBPR	"	"				4×1
15	CLS2						眠る人向CLS
16	BCOL	カラー					
17	TTLPR	X	Y				10×2
18	KABE	"	"				1×1
19	FUN1	"	"				} 同一ルーチン1×1
1A	FUN2	"	"				
1B	FAB2PR	"	"				
1C	UFB2PR	"	"				2×1
							4×1

## あとがき

①私はオール・マシン語のゲームばかり作っていますが、何を隠そうBASICのリンク方法がわからないだけの理由です。

表4 \$7F00~のワーク・エリアの詳細

アドレス	バイト数	内 容
7F00	2	MISSILEの座標
7F02	2	MY CANNONの座標
7F04	2	BASE UFOの座標
7F06	2	ミステリーUFOの座標
7F08	2	現在のスコア
7F0A	2	HI-SCORE
7F0C	1	MY CANNONの数
7F0D	2	乱数ワーク
7F12	1	衝撃波の出ている時間
7F13	1	衝撃波(FA)の幅(半径)
7F14	1	MY CANNONフラグ 5:異常 0:正常
7F16	1	UFOフラグ 1:異常 0:正常
7F23	1	MY CANNONの移動速度
7F24	1	UFOの出る速度
7F25	1	MISSILEの上昇速度
7F27	1	FALL ALIENの移動速度
7F2C	1	MISSILEを何発撃ったか
7F2E	1	UFOの降下速度
7F32	1	FALL ALIENの出てくる速度
7F39	1	面数

※上記ワーク・エリアは、プログラムの途中で書き込み、書き換えを行っているのですが、マニュアルでこの値を変えても何にもなりません。変更するにはサブルーチンINIT0、POWERUPを参照、変更してください(他のルーチンではSCOREなどを除いて変更、書き込みはしていません)。

ただし、MY CANONの数は\$574E。

# ALIEN FALL

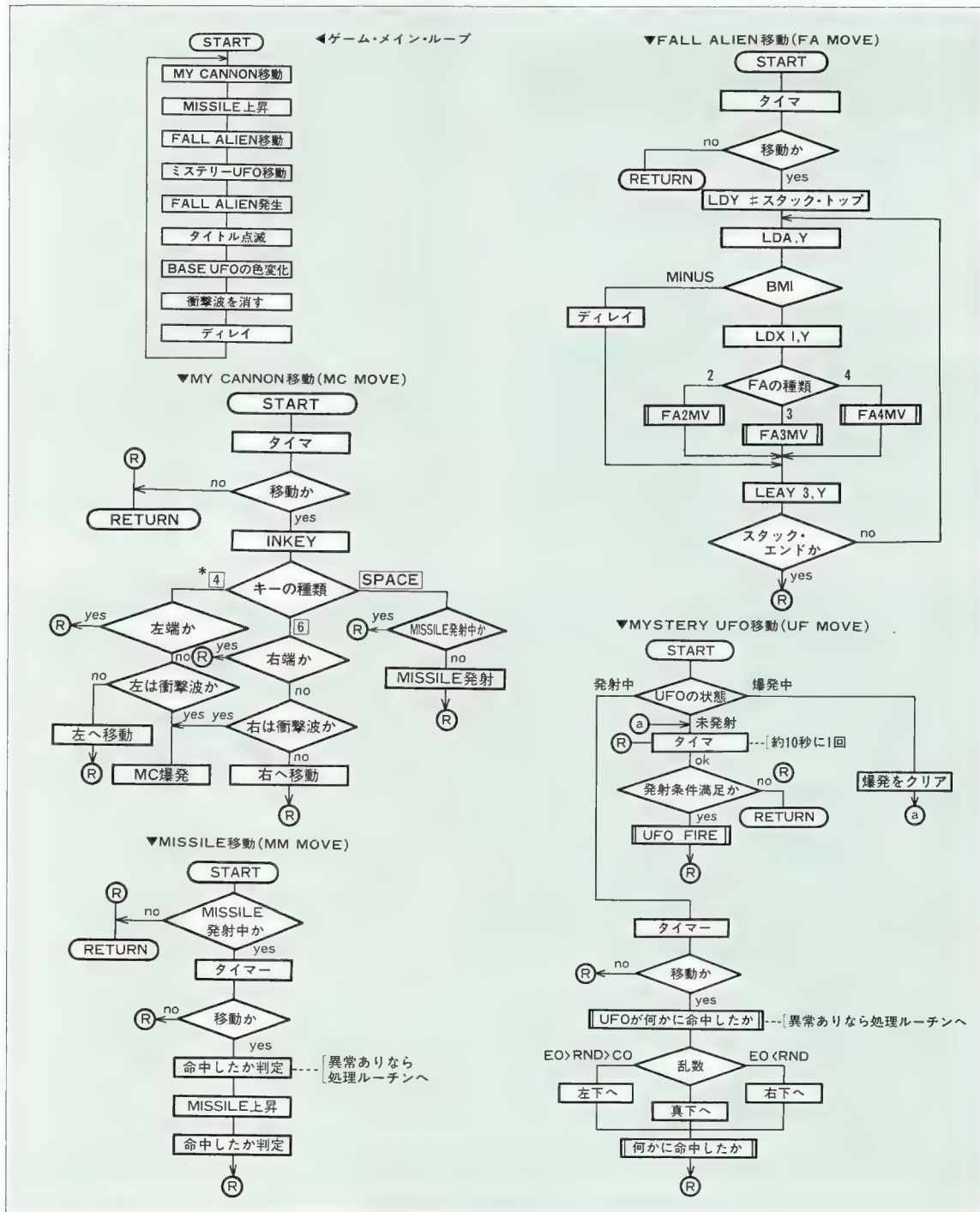
② 高畑氏がゲームはサブ側で、メイン側は音出し専門とありますが、私の場合、並列処理によるスピードと画面のデータの増加によるサブ側プログラム・エリアの減少を考えて、それには少し無理があるのではと思います。音出しについてもPSGが一般化してきたいま、問題はない構造の簡易化という点から見ても、やはりメイン処理はメイン側で行ない、サブ側はインテリジェント端末として使った方がよいのではないのでしょうか。

③ 解説派の方のお楽しみとして、隠しコマンドが入れています。名付けて闇夜のALIEN FALLと地中のALIEN FALL、その他です。探してみてください。

④ 筆者のハイ・スコアは36,150点です。

⑤ 書き忘れてました。スタート番地は&H4000です。

フローチャート



## ALIEN FALL ダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2D00	34	76	81	01	27	1E	02	27	22	81	03	27	26	81	04	97	
2D10	27	2A	81	05	27	2E	81	06	32	81	07	27	36	81	08	7A	
2D20	27	3A	35	F6	30	BD	00	61	BD	3A	35	F6	30	BD	00	67	C0
2D30	BD	32	35	F6	30	BD	00	6D	BD	3A	35	F6	30	BD	00	73	26
2D40	BD	22	35	F6	30	BD	00	79	BD	1A	35	F6	30	BD	00	7F	1E
2D50	BD	12	35	F6	30	BD	00	85	BD	0A	35	F6	30	BD	00	BB	14
2D60	BD	02	35	F6	3A	16	4F	8E	8D	BD	07	4C	81	0E	25	F7	44
2D70	35	96	34	12	BE	FD	0A	A7	01	86	03	A7	84	6F	84	E7	FD
2D80	01	86	02	A7	84	6F	84	35	92	00	0A	00	05	55	03	E1	F3
2D90	F0	10	10	00	40	3B	00	4D	00	00	00	00	00	0F	F5	10	F9
2DA0	10	00	C0	03	03	33	01	9A	00	00	00	00	FC	10	10	00	BD
2DB0	C0	03	00	00	0A	00	05	55	03	00	F8	10	10	10	C0	12	24
2DC0	00	00	03	4D	00	08	00	00	F8	10	10	10	60	09	00	00	E9
2DD0	05	00	05	00	05	00	F8	10	10	10	60	09	00	4D	00	26	13
2DE0	00	33	00	00	F8	10	10	06	00	03	0E	80	01	00	00	00	E3
2DF0	00	00	FE	10	00	C0	05	04	00	00	00	00	00	00	00	00	D7

Sum: B1 A4 17 FD 9B 85 B0 ED A4 12 60 7E 85 E7 73 34 :CD

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2E00	16	02	28	16	03	19	16	03	FC	16	04	AF	16	04	F2	16	72
2E10	05	6B	7E	39	FE	7E	3B	3F	7E	36	CA	7E	3B	0D	7E	3B	15
2E20	8B	7E	39	7E	57	1A	7E	3B	9E	7E	3A	6F	7E	3A	2F	EF	
2E30	7E	35	A7	7E	50	72	7E	3A	49	7E	35	1A	7E	3A	0F	7E	CB
2E40	39	7C	7E	39	BC	7E	39	CD	7E	39	A7	7E	39	AE	7E	39	6D
2E50	07	7E	39	85	7E	39	E1	7E	2D	64	3C	3D	3D	20	42	3F	
2E60	41	53	45	20	55	46	4F	00	3C	3D	3D	3D	20	4D	59	20	BC
2E70	43	41	4E	4F	4E	00	3C	3D	3D	3D	20	46	41	4C	4D	49	DE
2E80	53	49	4C	4E	45	00	3C	3D	3D	3D	20	46	41	4C	4D	20	D2
2E90	41	4C	49	45	4E	00	3C	3D	3D	3D	20	4D	59	53	54	45	0E
2EA0	52	59	20	55	46	4F	2B	43	61	75	74	69	6F	6E	20	21	F1
2EB0	21	29	00	3C	3D	3D	3D	20	CC	BD	BC	AC	00	3C	3D	3D	24
2ED0	3D	20	54	49	54	54	4C	45	00	22	41	4C	49	45	4E	20	DE
2ED0	46	41	4C	4C	22	00	C4	20	B6	82	C3	B1	D9	A1	00	2A	A5
2EE0	2A	2A	2A	2A	20	20	41	20	4C	20	49	20	45	20	4E	20	F1
2EF0	20	46	20	41	20	4C	20	4C	20	20	2A	2A	2A	2A	2A	00	B1

Sum: BC E4 6C 7A 79 A9 D9 30 EB 5D C6 B5 C7 9A C0 0F :A1

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
2F00	2A	2A	2A	2A	2A	20	50	4C	41	59	20	4D	41	4E	55	99	
2F10	41	4C	20	56	6F	6C	2E	20	31	20	2A	2A	2A	2A	2A	6F	
2F20	00	2A	2A	2A	2A	2A	20	20	43	6F	70	79	72	69	67	68	57
2F30	7A	20	2B	43	29	20	31	39	3B	33	20	2A	2A	2A	2A	05	
2F40	2A	00	2A	2A	2A	2A	2A	20	20	42	79	20	4E	2E	4D	61	74
2F50	73	61	6F	6B	61	20	4A	52	34	4F	46	2E	2A	2A	2A	79	
2F60	2A	2A	00	5B	43	41	53	54	5D	20	20	20	20	20	20	17	
2F70	20	20	5B	4E	41	4D	45	5D	20	20	20	20	20	20	20	19	
2F80	20	5B	50	4D	4F	49	4E	54	5D	00	42	41	53	45	20	55	46
2F90	4F	00	4D	59	53	54	45	52	59	20	55	46	4F	20	20	20	54
2FA0	20	3F	3F	3F	3F	00	46	41	4C	4C	20	41	4C	49	45	4E	C4
2FB0	20	20	20	20	32	30	2B	34	30	29	00	4D	59	20	43	C0	
2FC0	41	4E	4F	4E	00	5B	4D	49	53	53	49	4F	4E	5D	00	44	4A
2FD0	45	53	54	52	4F	50	55	46	4F	20	26	20	41	4C	49	2C	
2FE0	45	4E	20	21	21	00	5B	4B	65	79	20	46	75	6E	63	74	99
2FF0	69	6F	6E	5D	00	4D	49	53	53	49	4C	45	3A	5B	53	50	F1

Sum: A9 B3 BD 50 66 B3 CB 40 F3 16 BD 63 0B DF 9C 24 :00

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3000	41	43	45	5D	00	4D	4F	56	45	34	3C	E0	4D	4E	E0	87	
3010	E0	3E	3A	00	50	55	53	4B	20	27	20	20	4B	45	59	3B	
3020	20	5A	4B	45	4E	20	BE	C2	D2	B2	00	34	76	B6	3C	97	96
3030	7F	32	B6	0A	B7	7F	27	B6	0A	B7	7F	0C	BE	15	1B	BF	E4
3040	7F	02	BE	00	00	17	FD	F1	17	FD	EE	17	FD	D6	17	FD	14
3050	DC	BE	13	00	BE	7F	0A	17	01	BD	BE	00	00	17	FD	D9	2F
3060	17	FD	BE	7F	00	00	17	FD	C7	17	FD	C7	BE	00	64	17	2B
3070	FD	C7	7A	00	00	26	EF	BE	17	03	17	FD	9B	BE	15	0F	5C
3080	BF	7F	06	17	FD	BE	15	1B	BF	7F	02	17	FD	92	BE	16	
3090	16	14	BF	7F	00	17	FD	B3	17	FD	B3	BE	00	00	17	FD	3B
30A0	9B	17	FD	95	31	BD	FD	B2	BE	1E	01	86	07	17	FD	77	75
30B0	31	BD	FD	B4	BE	19	1B	96	07	17	FD	B3	31	BD	FD	B6	AB
30C0	BE	1B	14	86	07	17	FD	5F	31	BD	FD	E7	BE	1B	15	86	9D
30D0	07	17	FD	53	BE	1A	0F	31	BD	FD	BB	86	07	17	FD	47	B3
30E0	8E	1A	03	31	BD	FD	9F	86	07	17	FD	3B	BE	3D	02	31	FD
30F0	BD	FD	CA	BE	07	17	FD	2F	30	01	31	BD	FD	CB	86	07	6B

Sum: 7D DB D7 9A F9 BE D6 BE EA 51 59 34 99 19 43 6B :D6

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3100	17	FD	24	30	01	31	BD	FD	CD	B6	07	17	FD	19	BE	00	39
3110	00	17	FD	25	17	FD	2D	17	FD	1F	17	FD	1C	35	F6	34	31
3120	76	17	FD	12	BE	00	00	17	FD	0F	31	BD	FD	B1	BE	05	4C
3130	00	B6	04	17	FC	F1	31	BD	FD	C6	30	01	17	FC	EB	31	6C
3140	BD	FD	DE	30	01	17	FC	DF	31	BD	FD	F6	30	01	17	FC	80
3150	D6	31	BD	FE	0E	8E	05	B6	07	17	FC	CA	31	BD	FE	5E	
3160	2B	8E	14	07	86	06	17	FC	BE	31	BD	FE	25	BE	14	09	BA
3170	17	FC	B4	31	BD	FE	2F	BE	14	0B	17	FC	CA	31	BD	FE	DB
3180	3B	8E	14	0D	17	FC	CA	31	BD	FE	3A	BE	05	0F	86	07	C2
3190	17	FC	94	31	BD	FE	3B	8E	05	10	86	05	17	FC	8B	31	95
31A0	BD	FE	43	8E	05	12	86	07	17	FC	7C	31	BD	FE	46	BE	1F
31B0	05	13	86	03	17	FC	70	31	BD	FE	4A	BE	05	14	17	FC	E4
31C0	66	BE	14	BF	7F	02	17	FC	57	BE	05	06	BF	7F	04	9A	
31D0	17	FC	3F	BE	05	09	BF	7F	06	17	FC	39	BE	05	09	17	33
31E0	FC	3A	BE	05	0D	BF	7F	02	17	FC	36	31	BD	FE	25	BE	CA
31F0	05	17	B6	07	17	FC	30	BE	00	00	17	FC	3C	17	FC	39	15

Sum: 91 DB 26 61 6C 13 65 43 9C BC 94 4B 01 E2 55 0F :9B

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3200	17	FC	36	35	F6	B6	07	BD	2D	00	B6	7F	0D	2B	0A	BD	EF
3210	26	30	BD	01	70	17	FC	3F	39	BD	08	30	BD	01	66	17	AF
3220	FC	35	39	B6	45	5F	1F	01	BF	7F	04	31	BD	02	38	BE	7C
3230	07	6C	17	00	33	20	00	34	16	B6	45	5F	1F	01	BF	7F	AF
3240	04	17	FB	CE	34	10	BE	09	C4	17	FB	ED	35	10	17	FB	D9
3250	EB	30	89	FF	00	BF	7F	04	BC	12	00	24	E4	0E	13	00	2C
3260	BF	7F	04	17	FB	AC	35	96	A6	00	27	4E	81	01	27	1E	4D
3270	81	02	27	1F	81	03	27	20	B1	04	27	21	B1	06	27	22	31
3280	B1	07	27	B1	03	08	27	24	B1	09	27	25	20	DA	17	FB	8B
3290	AE	20	12	17	FB	AC	20	1C	17	FB	AA	20	17	17	FB	AB	96
32A0	20	12	17	FB	A6	20	0D	17	FB	AA	20	08	17	FB	A2	20	C9
32B0	03	17	FB	A0	17	DB	82	16	FF	AE	39	34	22	86	07	BD	E5
32C0	2D	00	8E	07	F0	31	BD	00	CA	17	FF	9C	31	BD	00	CD	C7
32D0	17	FF	95	31	BD	00	F9	17	FF	8E	31	BD	00	D7	17	FF	B1
32E0	87	31	8D	00	FB	17	FF	80	31	BD	00	FC	17	FF	79	8E	AA
32F0	00	00	17	FB	44	30	BD	00	8C	17	FB	5B	17	FB	3D	35	90



# ALIEN FALL

ダン・リポート

36E0	20	F0	34	02	17	FE	67	B1	03	26	0A	BD	EC	30	01	BD	:AD
36F0	D9	1C	FE	35	B2	1A	01	35	B2	34	10	17	FF	DB	30	1F	:00
Sum: 46 08 61 24 30 FB 1A 8D 60 3D A5 25 10 CC 01 9B :B1																	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3700	34	10	17	FF	DD	35	10	25	04	+9	30	1F	20	F3	35	90	:00
3710	16	17	FE	B2	A6	8D	00	E2	1F	8F	17	FE	4A	ED	B4	35	:6F
3720	96	34	12	BE	7C	B0	A6	84	27	0F	B1	FF	27	02	6A	B4	:8D
3730	30	01	8C	7C	CF	23	EF	35	92	34	06	1F	10	C0	A0	17	:C1
3740	04	10	35	06	20	EA	34	10	BE	7C	B0	6F	B4	63	B0	BC	:B9
3750	7C	CF	23	F7	35	90	BD	B1	30	8F	FF	00	16	FF	6B	17	:87
3760	FF	77	30	B9	01	00	16	FF	61	17	FF	6D	30	89	FF	01	:E2
3770	16	FF	57	17	FF	63	30	B9	01	01	16	FF	4D	8E	7C	C0	:CC
3780	E6	B0	C1	FF	27	08	BC	7C	0D	25	F5	BE	00	00	30	1F	:24
3790	1F	10	C0	A0	1F	98	39	BE	7C	B0	E6	80	C1	FF	27	08	:8E
37A0	BC	7C	C0	25	F5	BE	00	00	30	1F	10	C0	A0	1F	98	05	:05
37B0	39	34	16	34	10	BD	EC	E7	8D	00	3F	B6	7F	12	87	A7	:3F
37C0	35	10	1F	98	17	02	FC	17	FF	45	35	96	34	16	BD	C7	:D5
37D0	E7	8D	00	26	B6	7F	12	A7	BE	7F	A6	80	1F	98	17	02	:1F
37E0	D4	17	00	02	35	96	34	16	17	00	87	A6	80	00	0B	1F	:03
37F0	B9	17	FD	73	ED	B1	ED	B4	35	96	20	BD	53	7A	7F	06	:B9
Sum: EE BC 05 53 5D C5 66 22 D4 A6 15 BA BE 36 CF 99 :51																	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3800	7C	7F	07	20	3A	8D	49	7C	7F	07	20	33	8D	42	7C	7F	:51
3810	06	7C	7F	07	20	29	8D	3F	7F	07	20	22	8D	31	7C	7F	:91
3820	7A	07	20	1B	8D	2A	7A	7F	07	20	11	BD	20	11	BD	20	:54
3830	7A	07	20	0A	8D	19	7C	7F	07	20	00	34	25	00	34	25	:A5
3840	10	BE	7F	06	BD	35	00	BE	7F	06	BD	35	00	BE	7F	06	:64
3850	34	10	BD	35	00	BE	7F	06	BD	35	00	BE	7F	06	BD	35	:69
3860	1F	34	10	BD	35	00	BE	7F	06	BD	35	00	BE	7F	06	BD	:44
3870	02	0E	34	10	BD	35	00	BE	7F	06	BD	35	00	BE	7F	06	:8C
3880	34	16	BE	7F	0D	1F	10	30	8B	30	8B	1F	10	1E	89	3A	:49
3890	1F	10	1E	89	C3	00	81	BD	7F	0D	35	96	34	36	E6	A0	:5E
38A0	27	09	17	01	CA	30	B9	01	00	20	F3	35	B6	34	36	3A	:68
38B0	04	E6	A0	17	01	B9	30	B9	01	00	35	04	5A	26	F0	35	:F3
38C0	B6	34	10	BD	35	00	BE	7F	06	BD	35	00	BE	7F	06	BD	:D2
38D0	17	FC	2D	B6	FC	B2	35	B4	34	16	BE	3C	0A	10	F6	7F	:0E
38E0	B6	07	17	01	06	BE	2E	12	30	B9	04	00	BD	32	4A	26	:62
38F0	F7	B6	07	BE	2E	12	30	B9	04	00	BD	16	4A	27	05	5A	:85
Sum: AB 62 0F BC A0 BF 4F B7 B2 BE BC A7 B4 76 B7 A3 :D1																	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3900	26	F4	35	96	1E	10	B6	2E	5C	1E	10	B6	07	5A	26	E6	:44
3910	35	96	34	10	17	FB	9F	BF	FC	B2	BE	FF	11	16	01	60	:C5
3920	34	10	BD	35	00	BE	7F	06	BD	35	00	BE	7F	06	BD	35	:5C
3930	BE	7F	06	BD	35	00	BE	7F	06	BD	35	00	BE	7F	06	BD	:8C
3940	96	34	16	BE	7F	02	BD	8D	4F	5F	20	EC	34	16	BE	7F	:C3
3950	02	BD	13	CC	06	06	20	E0	BD	E7	7A	7F	02	20	CF	BD	:65
3960	E0	7C	7F	02	20	CB	34	10	BD	35	00	BE	7F	06	BD	35	:C5
3970	13	16	01	0C	34	32	BE	3C	06	10	BE	7F	06	BD	35	00	:DB
3980	11	35	B2	34	32	BE	3C	05	10	BE	7F	06	BD	35	00	BD	:A0
3990	35	B2	34	10	BD	35	00	BE	7F	06	BD	35	00	BE	7F	06	:5C
39A0	B6	BE	FF	06	16	00	D9	BD	66	7A	7F	0A	20	51	BD	5F	:55
39B0	7C	7F	0A	20	4A	BD	58	7A	7F	05	20	43	BD	51	7C	7F	:88
39C0	05	20	3C	BD	4A	7A	7F	05	20	32	BD	40	7C	0D	7C	0D	:80
39D0	7F	0A	7C	7F	05	20	28	BD	36	7A	7F	0A	7A	7F	05	20	:A9
39E0	1E	BD	2C	7C	7F	0A	7A	7F	05	20	14	34	10	BD	35	00	:3E
39F0	BF	FC	B2	F7	FC	B5	B7	FC	B6	BE	FF	07	16	00	B1	34	:4D
Sum: B1 0D 20 E9 04 0B 24 4F CA BC F9 A0 D5 9D 49 D5 :98																	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3A00	10	BD	35	00	BE	7F	06	BD	35	00	BE	7F	06	BD	35	00	:E1
3A10	10	BD	35	00	BE	7F	06	BD	35	00	BE	7F	06	BD	35	00	:D2
3A20	10	BD	35	00	BE	7F	06	BD	35	00	BE	7F	06	BD	35	00	:43
3A30	10	BD	35	00	BE	7F	06	BD	35	00	BE	7F	06	BD	35	00	:A3
3A40	10	BE	FF	01	BD	35	00	20	37	34	10	BE	FF	02	20	F4	:CE
3A50	34	10	BE	FF	15	20	ED	34	10	BE	7B	00	6F	B0	BC	7C	:3A
3A60	AF	23	F9	35	90	34	10	BD	35	00	BE	FF	03	20	11	3A	:CE
3A70	10	BD	35	00	BE	7F	06	BD	35	00	BE	7F	06	BD	35	00	:B1
3A80	BF	FC	B0	BD	40	BC	35	90	34	10	BD	35	00	BE	FF	03	:2C
3A90	BE	FF	08	20	EB	34	10	BD	35	00	BE	7F	06	BD	35	00	:AE
3AA0	20	DE	34	36	1F	12	BD	24	1E	12	BD	20	1E	12	31	A9	:31
3AB0	01	00	30	89	FF	00	5A	26	ED	35	B6	34	04	C6	32	BD	:CE
3AC0	E1	35	94	34	04	F6	7F	13	BD	35	B6	34	04	C6	32	BD	:10
3AD0	4D	27	38	2B	36	B1	2F	24	32	1E	10	34	10	17	FA	B7	:1D
3AE0	34	04	E6	B6	BD	F7	12	A7	B4	1F	98	35	04	35	10	44	:F5
3AF0	B1	05	26	06	17	FE	55	B3	7F	14	BD	35	00	10	BE	FF	:A4
Sum: 94 B0 43 BA F9 EB B7 9B C1 33 DD BF 5A 36 59 ED :AD																	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3B00	09	BF	FC	B2	10	BF	FC	B0	8D	A0	BC	35	B6	34	10	BE	:37
3B10	7F	00	BD	35	00	BE	7F	06	BD	35	00	BE	7F	06	BD	35	:00
3B20	BE	7F	00	17	F9	DA	BF	FC	B2	BE	FF	08	16	FF	51	34	:93
3B30	10	BE	7F	02	30	B9	0F	BF	7F	00	BD	35	00	35	90	34	:9B
3B40	10	17	F9	B2	BE	7F	00	30	01	BF	FC	B2	BE	FF	08	16	:32
3B50	FF	2E	34	32	34	04	F6	7F	13	BD	35	B6	34	04	C6	32	:A3
3B60	26	21	63	8E	00	1B	BC	2E	1B	22	17	1F	12	17	F9	01	:90
3B70	F6	A6	84	1F	21	34	04	A1	E0	26	02	BD	08	30	B9	01	:90
3B80	00	20	E4	35	B2	34	32	BD	35	00	BE	FF	08	16	FF	51	:F6
3B90	FC	B2	10	BF	FC	B0	8D	40	BC	17	F9	DA	BF	FC	B2	BE	:32
3BA0	17	FF	7B	8D	9A	7A	7F	01	BD	11	16	FF	40	34	32	BD	:BE
3BB0	35	00	10	BE	FF	1B	86	08	16	FF	DA	34	10	BE	7F	00	:E2
3BC0	30	01	BD	35	00	BE	7F	06	BD	35	00	BE	7F	06	BD	35	:10

3BD0	BE	77	FF	30	01	BC	7C	AF	22	0B	A1	B4	26	F5	6F	B4	:49
3BE0	20	F1	35	90	34	10	BE	77	FF	30	01	BC	7C	AF	22	F2	:1A
3BF0	A1	B4	27	F5	6F	B4	20	F1	34	16	F6	7F	0F	BE	17	05	:BD

# ALIEN FALL

ダンプ・リスト

```
40C0 B7 FD 05 A7 BC F5 1C AF 35 02 BD 02 BD 00 39 12 :4A
40D0 12 12 3D 3D 3D 3D 10 CE CF FF 7F D3 B0 7F D3 B0 :68
40E0 B6 D4 0A B6 D3 B0 27 F8 B7 D4 0A B6 D4 0A B6 D3 :6F
40F0 B1 4A C6 03 3D 30 BD 00 05 AD BB 16 FF BD 16 00 :CE
Sum: 01 65 B2 F0 ED BE 42 61 BB 64 72 BD 03 0F 72 7A :E2
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4100 51 16 00 67 16 01 38 16 01 5E 16 02 4F 16 02 F0 :01
4110 16 02 FF 16 01 CA 16 02 C4 16 02 AB 16 03 23 16 :E9
4120 03 2F 16 04 B2 16 01 FB 16 04 1E 16 04 21 16 03 :6C
4130 7B 16 03 B1 16 03 84 16 04 77 16 00 62 16 00 BF :90
4140 16 04 00 16 02 9D 16 02 A1 16 02 9E 16 02 FC 16 :68
4150 01 DF B7 D4 08 B6 D4 09 4F 5F BE 00 00 ED B1 BC :3C
4160 C0 00 25 F9 B6 D4 08 B7 D4 09 39 34 72 B6 D4 09 :76
4170 BD 02 20 26 B6 00 00 8D 0B 0E 00 8D 06 BE 80 :6A
4180 00 BD 01 39 33 B9 40 00 0F 3E 6F B1 7D F0 00 6F :E0
4190 21 31 3E 34 40 AC E1 25 F1 39 7D 19 09 35 F2 34 :CF
41A0 76 B6 D4 09 CC 00 00 ED BC 17 BD 19 CC 40 00 ED :04
41B0 BC 0F 8D 11 CC 80 00 00 ED BC 07 BD 09 B7 D4 09 :36
41C0 F6 FD 05 A7 BC CE 00 00 ED BC F6 C3 BE 80 ED BC :61
41D0 22 B6 C8 34 02 C6 50 3D E3 BC E6 1F 01 31 88 50 :77
41E0 10 AF BC DF EC BA EF B1 ED 89 00 9E AC BC D4 25 :4F
41F0 F3 10 BC 3E B0 25 E6 35 02 4A B1 FF 26 D5 39 34 :C1
Sum: B7 07 99 BA 02 FD 0B 6A 94 6F F2 BB FA 46 97 ED :69
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4200 06 B6 D4 09 F6 D3 B2 54 24 05 BE 00 00 BD 1B 54 :EB
4210 24 05 BE 40 00 BD 13 54 24 05 BE 80 00 BD 0B 87 :71
4220 D4 09 3D 3D 3D 3D 35 B6 16 03 34 30 31 89 40 00 :03
4230 10 AF BC F4 63 B0 63 B0 AC BC ED 25 F7 35 B0 34 :5F
4240 76 B6 D4 09 3D BE 01 E0 10 AE 89 B0 00 EE 89 40 :33
4250 00 EF B9 80 00 EE BA EF B9 40 10 AF B1 BC 03 :F1
4260 E8 25 E5 B7 D4 09 35 F6 B6 D4 01 B7 D3 B2 39 BD :0E
4270 01 39 33 89 00 00 27 10 D3 E8 00 64 00 0A 00 01 :87
4280 34 76 33 BD FF FA 31 BD FF E5 1F 10 30 BD FF E6 :D6
4290 6F BD FF E1 A3 B4 25 06 6C BD FF D9 20 F6 BD 0A :AC
42A0 E3 B1 34 40 AC E1 25 BE 35 F6 34 02 B6 BD FF C5 :CA
42B0 BB 30 A7 A0 35 B2 EC BC 34 06 34 02 B6 A0 3B 58 :5C
42C0 49 58 49 1F 01 35 04 3A 35 B6 34 10 BE 00 00 4D :57
42D0 27 0A BE 40 00 B1 01 27 03 BE 80 00 1F 10 35 90 :AD
42E0 BE D3 B2 4F BD 08 B6 01 BD 0A B6 02 B0 34 16 01 :01
42F0 34 02 1F 10 BD C2 35 02 BD D0 30 BB B6 D4 09 3D :D3
Sum: E0 61 25 4F 45 03 35 EE B2 99 B7 0A 99 67 9E 4D :F7
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4300 4F A7 B4 A7 88 50 A7 B9 00 A0 A7 B9 00 F0 A7 B9 :19
4310 01 40 A7 B9 01 90 A7 B9 01 E0 A7 B9 02 30 B7 D4 :00
4320 09 35 96 34 30 31 BD 03 D7 BE D3 B2 17 02 BB 35 :B9
4330 B0 34 30 31 BD 09 91 20 F0 34 06 B1 30 20 13 B1 :1B
4340 39 22 0F C6 08 3D 30 BD 08 BE 34 10 E3 E1 1F 01 :F0
4350 35 B6 C6 08 3D C3 DB 00 1F 01 35 B6 34 32 B6 D3 :2B
4360 B4 BD D6 1F 12 BE D3 B2 F6 D3 B6 54 24 03 4F BD :D1
4370 10 54 24 04 B6 01 BD 09 54 24 04 B6 02 BD 02 35 :71
4380 B2 34 36 34 02 1F 10 17 FF 2E 35 02 17 FF 38 30 :7D
4390 BB B6 D4 09 3D A6 A0 A7 B4 A6 A0 A7 B8 50 A6 A0 :D7
43A0 A7 B9 00 A0 A6 A0 A7 B9 00 F0 A6 A0 A7 B9 01 40 :ED
43B0 A6 A0 A7 B9 01 90 A6 A0 A7 B9 01 E0 A6 A0 A7 B9 :D4
43C0 02 30 B7 D4 09 35 B6 31 BD 05 15 BE D3 B2 4F BD :78
43D0 B0 B6 01 31 2B BD AA B6 02 31 2B BD 34 31 BD 05 :2F
43E0 17 20 B8 31 BD 08 99 20 E2 00 68 AC FC 12 12 12 :A1
43F0 12 12 2B 06 31 BD 09 30 20 D1 31 BD 03 08 20 CB :F1
Sum: 70 D4 3C 2B F8 25 D3 3B F4 4C 67 A5 EB 2A B6 B1 :98
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4400 BE D3 B4 17 FE 7A BE D3 B2 31 BD FE 62 B6 D3 B6 :E4
4410 20 17 7F D3 B4 BE D3 B4 17 FE 65 BE D3 B2 31 BD :6D
4420 FE 4D 41 23 B6 D3 B6 20 00 34 24 BF D3 B2 B7 D3 :C4
4430 B6 E6 A0 27 08 F7 D3 B4 17 FF 21 7C D3 B2 20 F1 :A5
4440 35 A4 31 BD 0A CA BE D3 B2 20 0C 31 BD 08 49 20 :D3
4450 F5 31 BD 02 AB 20 EF 34 22 4F BD 0A B6 01 BD 06 :C5
4460 B6 02 BD 02 35 A2 34 16 34 02 1F 10 17 FE 49 35 :30
4470 02 17 FE 56 30 BB B6 D4 09 3D EC A1 ED B4 EC A1 :B3
4480 ED BB 50 EC A1 ED B9 00 A0 EC A1 ED B9 00 F0 EC :47
4490 A1 ED B9 01 40 EC A1 ED B9 01 90 EC A1 ED B9 01 :F0
44A0 E0 EC A1 ED B9 02 30 B7 D4 09 35 31 BD 0A 90 :C6
44B0 BE D3 B2 20 0C 31 BD 02 47 20 F5 31 BD 06 A9 20 :E8
44C0 EF 34 22 4F BD 0A B6 01 BD 06 B6 02 BD 02 35 A2 :33
44D0 34 16 34 02 1F 10 17 FD FC 35 02 17 FD EC 30 BB :94
44E0 B6 D4 09 3D EC A1 ED B4 A6 A0 A7 02 EC A1 ED B8 :BF
44F0 50 A6 A0 A7 B8 52 EC A1 ED B9 00 A0 A6 A0 A7 B9 :30
Sum: 69 03 1B 4A ED 32 DE B5 D4 BA 65 3E F6 76 05 AE :A0
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4500 00 A2 EC A1 ED B9 00 F0 A6 A0 A7 B9 00 F2 EC A1 :8A
4510 ED B9 01 40 A6 A0 A7 B9 01 42 EC A1 ED B9 01 90 :04
4520 A6 A0 A7 B9 01 92 EC A1 ED B9 01 E0 A6 A0 A7 B9 :63
4530 01 E2 EC A1 ED B9 02 30 A6 A0 A7 B9 02 32 B7 D4 :4D
4540 09 35 96 31 BD 08 61 20 0A 31 BD 0A 3B 20 04 31 :77
4550 BD 01 AD BE D3 B2 34 22 4F BD 0A B6 01 BD 06 B6 :2A
4560 02 BD 02 35 A2 34 52 CE 00 00 C6 10 BD 08 33 CB :22
4570 50 5A 26 F8 35 D2 34 16 34 02 1F 10 17 FD 39 35 :00
4580 02 17 FD 46 30 BB 1F 30 30 BB B6 D4 09 3D EC A1 :7E
4590 ED B4 EC A1 ED 02 EC A1 ED 04 EC A1 ED 06 EC A1 :78
45A0 ED 08 B7 D4 09 35 96 31 BD 06 05 BE D3 B2 BD 07 :C4
```

```
45B0 39 31 BD 06 5B 20 F4 34 22 4F 8D 0A B6 01 BD 06 :C2
45C0 B6 02 BD 02 35 A2 34 16 34 02 1F 10 17 FC E9 35 :CE
45D0 02 17 FC F6 30 BB B6 D4 09 3D EC A1 ED B4 EC A1 :21
45E0 ED 02 EC A1 ED 88 50 EC A1 ED 88 52 EC A1 ED B9 :98
45F0 00 A0 EC A1 ED B9 00 A2 EC A1 ED B9 00 F0 EC A1 :C5
Sum: 06 59 79 22 78 F4 7F 1E 5D 7C 6B 06 B4 D6 61 91 :C9
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4600 ED B9 00 F2 EC A1 ED B9 01 40 EC A1 ED B9 01 42 :F2
4610 EC A1 ED B9 01 90 EC A1 ED B9 01 92 EC A1 ED B9 :2D
4620 01 E0 EC A1 ED B9 01 E2 EC A1 ED B9 02 30 EC A1 :89
4630 ED B9 02 32 B7 D4 09 35 96 A1 ED 02 EC A1 ED 04 :17
4640 EC A1 ED 06 EC A1 ED 08 B7 D4 09 35 96 31 BD 05 :24
4650 5F BE D3 B2 BD 07 39 31 BD 05 B5 20 F4 34 22 4F :70
4660 BD 0A B6 01 BD 06 B6 02 BD 02 35 A2 34 16 34 02 :1F
4670 1F 10 17 FD 31 35 02 17 FD 3E 30 BB B6 D4 09 3D :88
4680 EC A1 ED B4 EC A1 ED 02 EC A1 ED 88 50 EC A1 ED :46
4690 BB 52 EC A1 ED B9 00 A0 EC A1 ED B9 00 A2 EC A1 :AF
46A0 ED B9 00 F0 EC A1 ED B9 00 F2 EC A1 ED B9 01 40 :9F
46B0 EC A1 ED B9 01 42 EC A1 ED B9 01 90 EC A1 ED B9 :DD
46C0 01 92 EC A1 ED B9 01 E0 EC A1 ED B9 01 E2 EC A1 :EA
46D0 ED B9 02 30 EC A1 ED B9 02 32 B7 D4 09 35 96 00 :3E
46E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
46F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
Sum: F9 44 EC 43 67 AB 45 CB F1 B4 55 DF 6E 19 B0 FB :93
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4700 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4710 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4720 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4730 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4740 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4750 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4760 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4770 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4780 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4790 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
47A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
47B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
47C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
47D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
47E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
47F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
Sum: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4800 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4810 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4820 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4830 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4840 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4850 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4860 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4870 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4880 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
4890 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
48A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
48B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
48C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
48D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
48E0 1B 3C 7E 1B 1B BD 1B 91 00 00 1B 1B BD 1B BD :42
48F0 00 00 1B FF 1B 1B FF 08 00 00 00 FF 00 00 00 :4D
Sum: 1B 3C 96 17 30 D5 17 99 00 00 1B 1B 17 BD 1B BD :8F
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4900 99 99 FF 00 00 FF 99 99 00 00 99 FF FF 99 00 :92
4910 00 00 C0 03 FD BF 43 C2 7F FE 73 CE 61 B6 00 00 :29
4920 01 B0 C7 E3 C2 43 3C 3C 01 B0 73 CE 61 B6 01 B0 :D2
4930 FB 1F C0 03 FC 3F 7E 7F FE 03 C0 01 B0 FC 3F :0D
4940 00 1B 00 00 1B 00 00 3C 01 B0 3C 01 B0 7E 01 B1 :2A
4950 FF B1 D5 FF AB B1 66 B1 00 1B 00 1B 00 00 A5 :3C
4960 00 00 BD 00 00 99 00 00 24 00 2A 24 54 00 A5 :C1
4970 00 1B 00 B0 1B 01 00 3C 00 00 24 00 00 66 00 01 :78
4980 FF B0 7F FF FE 01 66 B0 00 00 00 00 00 00 00 :E2
4990 00 00 00 00 00 7F FF FF FE 00 00 00 00 00 00 :FB
49A0 00 00 01 00 00 01 FF FF BE 3B 1C 71 FF FF B0 :D1
49B0 00 00 00 00 00 00 01 00 00 00 FF FF FF FF FF :7B
49C0 FF FF FF FF FF FF FF FF FF FF FF FF FF FF :1F
49D0 FD SA AA AB 56 BF 6C EF 1F 56 BD SA AA D5 A9 7D :4D
49E0 6A FB 01 FD 6F 5A B5 55 AA B6 BF 80 00 0F F5 76 :4A
49F0 AD SA BE AF F0 00 00 00 2A AA AA 55 55 54 00 00 :E0
Sum: A3 14 C0 3D 46 F3 03 CF 13 57 68 C9 1C 3E 36 0E :FB
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
4A00 00 00 00 7F FF FF FE 00 00 00 00 00 00 7B 00 :F3
4A10 1E 00 00 00 00 00 00 00 00 00 00 1F 00 00 :35
4A20 3F FF 00 00 FF FC 00 00 00 00 00 00 7F FE 00 :B6
4A30 00 00 00 00 00 7C FF FF FE 00 00 00 00 00 00 :FB
4A40 00 00 01 00 00 00 01 FF FF FF FF FF FF FF FF :7A
4A50 00 00 00 80 00 00 01 00 00 00 FF FF FF FF FF :7B
4A60 FF FF FF FF FF FF FF FF FF FF FF FF FF FF :68
4A70 AB 36 59 66 CD 95 31 57 1E C9 9A 6C F3 66 CF :98
4A80 93 7B 01 F3 3B 36 6C B3 6C DC CF B0 00 0F D9 BB :C9
```



# ALIEN FALL

## ダンプ・リスト

4A90	36	6C	DD	9B	F0	00	00	00	3F	FF	FF	FF	FF	FC	00	00	:41
4AA0	00	00	00	7F	FF	FE	00	00	00	00	00	00	00	7F	FF	FF	:F8
4AB0	FE	00	00	00	00	00	00	F8	00	00	00	00	00	00	00	00	:15
4AC0	3F	FF	00	00	FF	FC	00	00	00	00	00	00	00	7F	FE	00	:B6
4AD0	00	00	00	00	7F	FF	FE	00	00	00	00	00	00	00	00	FF	:7A
4AE0	FF	FF	FF	00	00	00	01	FF	80	00	00	00	01	FF	FF	B0	:FB
4AF0	00	00	00	FF	FF	FF	00	00	00	FF	FF	FF	FF	FF	FF	FF	:F6

Sum: 0C 16 36 70 F2 BD 9B F5 C2 22 A2 E7 ED 5F 8D 5C :A6

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4B00	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	F9	DB	:C6
4B10	3B	F1	C7	1E	3C	7C	DB	9F	1E	3C	7B	8F	3C	7B	F1	DE	:27
4B20	3C	7B	01	F0	F8	F1	E3	8F	1F	1F	0F	80	00	00	DE	33	:ED
4B30	C7	8F	1C	7B	F0	00	00	00	3F	FF	FF	FF	FF	FC	00	00	:14
4B40	00	00	00	7F	FF	FF	FE	00	00	00	00	00	00	7B	00	00	:F3
4B50	1E	00	00	00	00	00	00	F8	00	00	00	00	00	00	00	00	:35
4B60	3F	FF	00	00	FF	FC	00	00	01	C0	E6	46	11	5A	BE	:84	
4B70	EB	2B	E3	F8	3F	7D	FC	37	41	02	90	7E	22	41	80	1C	:2A
4B80	39	C7	6E	7F	FF	FF	FF	FF	FF	3F	FF	FF	FF	FF	FF	FF	:18
4B90	FF	FE	FF	FE	FE	FE	FE	FE	FE	FE	FE	FE	FE	FE	FE	FE	:7A
4BA0	99	2B	00	09	90	41	34	90	41	02	90	7F	E2	4F	87	1C	:F5
4BB0	80	00	00	00	01	C3	FF	FE	C3	E7	F2	4F	E7	FE	E2	4F	:3B
4BC0	7F	E2	4F	FE	E2	4F	E0	00	00	00	00	00	00	00	00	00	:CD
4BD0	80	00	00	00	01	C3	FF	FE	C3	E7	F1	8F	E7	FE	E1	8F	:7F
4BE0	FF	E1	8F	7F	7A	A1	85	FE	00	00	00	00	00	7F	FE	00	:DE
4BF0	00	00	00	00	00	00	00	00	00	03	EF	F7	C0	07	DF	FE	:6A

Sum: 3E D1 39 85 F4 BB CB 16 13 5E 13 52 CC 30 F4 :17

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4C00	00	00	1F	F8	00	00	1F	F8	00	00	00	00	00	00	00	00	:2E
4C10	3B	76	31	C3	47	FF	7F	F1	5F	FE	3F	FB	1F	F2	0F	FC	:08
4C20	08	04	01	04	5F	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	2F	:12
4C30	3C	77	B1	FB	7F	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	:90
4C40	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	:97
4C50	38	71	B1	C3	40	CE	51	41	C3	CA	8B	0F	0F	F6	04	:31	
4C60	DF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	:06
4C70	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
4C80	DB	DB	DB	DB	AS	AS	AS	AS	AS	AS	AS	AS	AS	AS	AS	AS	:00
4C90	E7	E7	E7	E7	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	C3	:63
4CA0	0C	30	0F	F0	05	AB	00	00	00	00	00	00	00	00	00	00	:39
4CB0	DF	BF	AF	F5	AB	AB	D5	00	00	00	00	00	00	00	00	00	:7C
4CC0	0F	F0	0F	F0	00	00	00	00	00	00	00	00	00	00	00	00	:7F
4CD0	0A	00	C3	50	0D	9E	79	B0	00	00	00	00	00	00	00	00	:E2
4CE0	05	7E	7E	0A	0B	82	41	00	AB	FC	3F	D5	D7	80	01	EB	:30
4CF0	AB	00	00	D5	DE	00	00	7B	AE	00	00	75	DB	00	00	DB	:B2

Sum: 0B 9A 9A 9F 6A 5D 3D 4B 7C 87 80 9F 82 F4 5C B3 :0E

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4D00	AS	B0	01	AS	DB	FC	3F	DB	00	00	FF	FF	00	01	FF	FF	:39
4D10	03	FE	7F	00	07	E0	07	E0	07	E0	03	FE	7F	00	00	00	:1C
4D20	01	FE	80	0F	FF	FF	00	00	00	1B	42	66	3C	18	00	00	:91
4D30	3C	77	FF	FF	FF	FF	3C	1B	00	00	3C	1B	00	00	00	00	:19
4D40	00	3C	3C	3C	24	24	24	1B	7E	7E	FF	FF	FF	FF	7E	1B	:C6
4D50	00	1B	1B	1B	1B	1B	00	00	00	00	3C	42	42	42	3C	00	:CE
4D60	00	00	00	7F	80	00	00	00	00	00	3C	00	7F	80	00	00	:3A
4D70	00	3E	00	3C	00	3F	40	00	00	FF	A9	AD	B1	FF	00	00	:77
4D80	00	FF	08	7E	12	12	FF	00	10	10	7E	00	24	44	B2	00	:30
4D90	10	10	7E	10	10	10	00	00	00	24	24	43	00	00	00	00	:8C
4DA0	00	20	FC	24	24	24	C3	00	00	00	00	00	00	00	00	00	:4B
4DB0	00	00	01	CE	0F	7F	31	E0	0F	8C	7E	01	CC	06	61	:37	
4DC0	3B	E7	CF	8C	FC	FE	02	CC	06	3B	E7	CF	8C	FE	FC	:FC	
4DD0	02	CC	06	60	3B	E7	FF	4C	FC	FE	02	CC	06	60	34	:E7	
4DE0	FF	4C	FC	FE	04	CC	06	62	34	E7	BE	CC	FC	FE	04	:CC	
4DF0	06	7E	34	E0	3E	CC	FC	FE	07	CC	62	34	E7	BE	CC	:BC	

Sum: 34 39 5A 3D 68 25 F3 43 E1 5D C0 CB 1B 3B B1 76 :0D

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4E00	FC	FE	08	CC	06	60	32	E7	FD	CC	FC	FE	08	CC	06	:4A	
4E10	32	E7	FD	CC	FC	FE	08	CC	06	60	32	E7	FD	CC	FC	:F2	
4E20	10	CC	26	60	B2	E7	FB	CC	F4	F6	10	CC	26	60	B1	:E3	
4E30	FB	CC	E4	E6	39	EF	EF	7F	B9	E1	E1	B4	04	06	00	:30	
4E40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	
4E50	00	00	7F	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	:3C	
4E60	FF	DB	30	73	03	00	7F	FF	FF	FF	DB	10	B3	03	00	:96	
4E70	7F	FF	FF	FF	FF	DB	00	B3	03	00	7F	FF	FF	FF	DB	:5C	
4E80	00	B3	03	00	7F	FF	FF	FF	DB	41	33	03	00	00	7F	:FE	
4E90	FF	FF	FF	FF	FF	C1	33	03	00	7F	FF	FF	FF	DB	41	:5A	
4EA0	03	00	7F	FF	FF	FF	DB	02	33	03	00	7F	FF	FF	FF	:0A	
4EB0	FF	DB	02	33	03	00	7F	FF	FF	FF	DB	02	33	03	00	:9A	
4EC0	7F	FF	FF	FF	FF	DB	04	33	08	7F	FF	FF	FF	FF	DB	:F4	
4ED0	04	33	1B	18	7F	FF	FF	FF	DE	1E	7B	FB	FB	00	00	:4F	
4EE0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:FB	
4EF0	FF	FF	B1	CE	0F	7F	31	DF	F0	73	B3	B1	B1	CC	06	:06	

Sum: 3A 0F DB 45 BD 92 56 76 1A D6 B1 90 FA 7C 7A 45 :BA

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4F00	3B	DB	30	73	03	01	B2	CC	06	60	3B	DB	10	B3	03	01	:42
4F10	B2	CC	06	60	3B	DB	00	B3	03	01	B2	CC	06	60	34	DB	:3B
4F20	00	B3	03	01	B4	CC	06	62	34	DB	41	33	03	01	B4	CC	:43
4F30	06	7E	34	DF	C1	33	03	01	87	CC	06	62	34	DB	41	F3	:8A
4F40	03	01	B8	CC	06	60	32	DB	02	33	03	01	B8	CC	06	:8B	
4F50	32	DB	02	33	03	01	B8	CC	06	60	32	DB	02	33	03	:10	
4F60	90	CC	26	60	B2	DB	04	33	08	09	90	CC	26	61	B1	:C6	
4F70	04	33	1B	19	B9	EF	EF	7F	B9	DE	1E	7B	FB	FB	FF	:A3	

4F80	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	:F8
4F90	00	E7	E7	E7	E7	E7	E7	E7	E7	E7	E7	E7	E7	E7	E7	E7	:7C
4FA0	0C	30	0F	F0	05	AB	00	00	00	00	00	00	00	00	00	00	:39
4FB0	DF	BF	AF	F5	AB	AB	D5	00	00	00	00	00	00	00	00	00	:7C
4FC0	0F	F0	F0	F0	F0	F0	F0	F0	F0	F0	F0	F0	F0	F0	F0	F0	:7F
4FD0	0A	C0	C3	50	0D	9E	79	B0	00	00	00	00	00	00	00	00	:E2
4FE0	05	7E	7E	0A	0B	82	41	00	AB	FC	3F	D5	D7	80	01	EB	:3D
4FF0	AB	00	00	D5	DE	00	00	7B	AE	00	00	75	DB	00	00	DB	:B2

Sum: 3C EC 6C AB B6 2D 6A 9A DB E6 93 CB FB DE 01 DF :CB

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5000	20	1A	53	2B	AD	FF	50	02	4								



```

5470 8E 10 12 86 05 BD 39 EB C6 F0 8E 15 12 86 07 BD :D1
5480 3A 6F 31 BD FF BA 8E 17 12 86 06 BD 38 9C C6 3D :F7
5490 86 07 8E 1B 12 BD 3A 6F C6 64 B6 7F 0C 3D F3 7F :C8
54A0 08 FD 7F 08 BD 39 74 4F BD 36 AD C6 64 B6 7F 0C :5A
54B0 3D 1F 02 8E 1D 12 86 03 BD 39 92 BE 00 00 BD 53 :CA
54C0 1A BD 53 1A BD 50 0A 8E 00 00 BD 53 1A BD 56 E8 :0E
54D0 8E 00 00 BD 53 1A BD 50 10 8E 27 10 BD 53 1A 16 :DA
54E0 FB 4C 32 62 86 01 17 FB 3D 34 10 BE 00 00 BD 53 :86
54F0 1A BD 53 1A 35 10 86 FF BD 37 21 4A 26 FA 7A 7F :86
    
```

Sum: 33 14 67 B0 5A 57 B7 24 6F 83 B0 3F 60 DD 55 98 :C5

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
S500 0C 10 27 00 C6 BD 38 DB BD 3B 1E BD 3B 3F BD 38 :18
S510 50 BD 39 41 8E 00 00 BD 53 1A BD 53 1A BD 53 1A :93
S520 BD 57 26 10 8E 7C D0 6D A4 2B 05 AE 21 BD 36 D9 :00
S530 31 23 10 8C 7D FC 23 EF BD 53 E5 86 03 BD 3B CE :BF
S540 8A 80 BD 3B CE BD 39 2E 16 FA E3 2A 2A 2A 2A 20 :AF
S550 47 41 4D 45 20 4F 56 45 52 20 2A 2A 2A 2A 2A 2A :68
S560 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A 2A :A0
S570 2A 2A 00 2A 20 CB CE AE 2D C6 20 CB CF 20 3D :BB
S580 3E 33 00 2A 20 CC C2 B3 20 20 20 20 20 20 20 3D :19
S590 3E 32 00 2A 20 B2 BF B6 DE BC B2 20 20 20 20 3D :EA
S5A0 3E 31 00 2A 20 BE C2 B2 B2 20 20 20 20 20 20 3D :BA
S5B0 3E 30 00 50 55 53 4B 20 27 53 27 20 4B 45 59 20 :9B
S5C0 54 48 45 4E 20 53 54 41 52 54 00 34 02 86 0A BD :60
S5D0 50 19 35 02 8E 00 00 BD 53 1A BD 53 1A BD 53 1A :9D
S5E0 8E 00 00 BD 53 1A BD 53 1A BE 7F 08 BC 3C EE 23 :30
S5F0 05 BF 3C EE 20 12 BC 3C F1 23 05 BF 3C F1 20 08 :45
    
```

Sum: 9E 42 B0 7A 6D 44 FB 54 3B E2 1C B0 B1 DB 16 76 :A2

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
S600 BC 3C F4 23 03 BF 3C F4 31 BD FF 53 8E 0E 0B 86 :3E
S610 06 BD 38 9C 8E 0E 0D BD 38 9C 31 BD FF 2D 8E 0E :57
S620 0C 86 07 BD 38 9C 86 04 BE 10 10 31 BD FF 44 BD :20
S630 38 9C 30 01 31 BD FF 4B BD 38 9C 30 01 31 BD FF :8C
S640 52 BD 38 9C 31 BD FF 5B 30 01 BD 38 9C BE 00 00 :4B
S650 BD 53 1A BD 53 1A BD 3B C1 81 30 26 2B BD 50 16 :2F
S660 31 BD FF 4F 8E 05 17 86 07 BD 38 9C BD 3B C1 81 :0B
S670 53 27 2C 81 73 27 23 81 30 26 F1 17 E3 CB 17 F9 :81
S680 92 8E 05 17 20 BE 20 E4 81 31 27 0E 81 32 27 0F :EE
S690 81 33 27 10 81 2A 27 11 20 BC BD 3A 3F 20 17 BD :D4
S6A0 3A 49 20 12 BD 3A 50 20 0D BD 38 80 B6 7F 0D 84 :64
S6B0 07 BD 3A 95 20 A0 16 F9 6F 7C 7F 36 10 26 F9 B1 :E2
S6C0 BD 3A 65 39 00 7C 7F 34 10 26 F9 A5 7C 7F 33 B6 :7C
    
```

```

56D0 7F 33 B1 01 10 23 F9 99 7F 7F 33 73 56 C4 2B 04 :E6
56E0 BD 3A 2F 39 BD 3A 1F 39 7F 7F 0F 7F 7F 2C BD 57 :F9
56F0 A6 BD 3A 57 BD 53 E5 BD 37 46 B6 01 BD 3A 95 BE :C4
    
```

Sum: 8C 0A B5 3E 87 B7 ED 6B 3E 66 4E E8 16 59 B6 80 :E6

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
S700 00 64 BD 53 1A 4C B1 08 25 F2 BD 3A 49 7F 7F 16 :CE
S710 7F 7F 14 8E 15 18 BF 7F 02 BE 00 00 BF 7F 00 BF :98
S720 7F 06 BD 57 3C 39 BE 15 18 BF 7F 02 BE 00 00 BF :56
S730 7F 00 BF 7F 06 7F 06 7F 16 7F 7F 14 39 BD 3A 57 :2D
S740 3A 3F BD 37 46 BD 53 E5 BD 3C 07 BD 3C F7 BD 38 :8D
S750 DB BD 3B FB BD 39 B3 BD 39 74 BD 39 2E 39 7F 7F :06
S760 0F BE 00 00 BF 7F 00 BF 7F 06 BF 7F 08 1F 10 86 :1A
S770 04 B7 7F 0C 86 05 B7 7F 12 86 0A B7 7F 13 F7 7F :68
S780 14 F7 7F 16 F7 7F 2C F7 7F 2F 86 0F B7 7F 30 7F :61
S790 7F 39 7C 7F 39 BD 0F 86 0F B7 7F 2D 8E 15 18 BF :FA
S7A0 7F 02 BD 57 3C 39 3A 06 B6 7F 39 B1 03 2A 8B 86 :0B
S7B0 1E B7 7F 23 86 5A B7 7F 32 86 2B B7 7F 27 86 0A :5A
S7C0 B7 7F 25 86 50 B7 7F 2E 86 14 B7 7F 24 86 14 B7 :DA
S7D0 7F 3B CC 00 28 FD 7F 3A 35 86 B1 05 24 17 86 19 :7C
S7E0 B7 7F 23 86 46 B7 7F 32 86 23 B7 7F 27 CC 00 1E :7D
S7F0 FD 7F 3A 35 86 B1 07 24 1C 86 37 B7 7F 32 86 18 :FF
    
```

Sum: BC CB 49 42 EF 21 B4 52 1B 28 69 CF F9 14 32 E4 :90

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
S800 B7 7F 27 86 0F B7 7F 24 CC 00 19 FD 7F 3A 86 14 :81
S810 B7 7F 23 35 86 B1 09 24 11 86 2D B7 7F 32 86 14 :88
S820 B7 7F 27 86 0A B7 7F 24 35 86 B1 08 24 08 CC 00 :86
S830 0F FD 7F 3A 35 86 B1 0F 24 12 CC 00 01 FD 7F 3A :C7
S840 86 28 B7 7F 32 86 32 B7 7F 2E 35 86 B6 1E B7 7F :C7
S850 32 86 0A B7 7F 27 B7 7F 23 86 28 B7 7F 2E 35 86 :45
S860 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
S870 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
S880 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
S890 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
S8A0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
S8B0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
S8C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
S8D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
S8E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
S8F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
    
```

Sum: EC 2B B1 B1 85 22 71 B1 DB D2 F0 FC 2B BD 43 67 :64

## R A N D O M

## B O X

### F-BASIC Ver.1.0をFM-7で

■大森 智

“FM8とFM7マシン語コンバータ”が発表されていますが、実は、そのようなものを打ち込まなくても…という話があるのです(ついでにマシン語コンバータでは、BASIC ROMのサブルーチンを使わなければマシン語コンパチと明記されていますが、サブ・システムのROMも違ってきます。よってここに発表するやりかたでも“FMラリー”などは走りません)。その方法は、ずばりFM-8のBASICをFM-7でというわけです。

安全のためにClear、& H9FFをしたあと下記のプログラムをFM-8で打ち込みEXEC & H7F00とし、SAVEM “CASO:FM8BASIC”、& HA00、& H7F40、& H7F20としてください。そこでFM-7でLOADM “CASO:”、Rとすれば走ります。

を基本として作ったので、ほとんど同じですがCLEARのときに“Out of Memory”にならないようにしました。

FM-7の裏RAMはRESETしても書きかわらないのでうっかりRESETしたとき&H7F20以下に

B7FD 0F 1A 50 4F 4D 8E 80 00 7E 89 12

を打ち込み&H7F20から実行してください。

注1. このプログラムは個人の機器で個人で行なってください。著作権問題がありますので悪しからず。

注2. カセットのボーレートがFM-7を倍速のままにしておくと約2400ボーとなります。私は、FMデータ・レコーダーを使っているため、2400ボーで、一部プログラムをSAVEしています。1600ボーとの間とは互換性はありません。

#### マシン語ダンプ・リスト

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
7F00 BE 0A 00 E6 B9 7E 00 E7 B0 8C 7E 00 26 F5 39 00 :4A
7F10 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
7F20 B7 FD 0F BE 0A 00 4F E6 B4 E7 B9 7E 00 A7 80 8C :B5
7F30 7E 00 26 F5 8E B8 00 7E B9 2D 00 00 00 00 00 00 :E1
Sum: C3 07 35 67 21 06 4F 4B BD A0 07 7E 26 9C B9 8C :E0
    
```



FM-7/8

Kコンパイラ用ゲーム

# キンチョマン



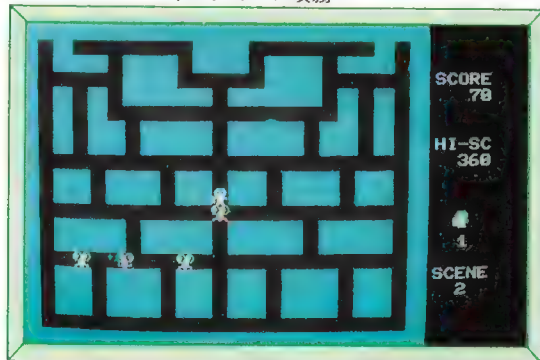
■服部浩憲

198X年、アメーバー状のインベダーが地球を襲ってきた。銃や刃物は彼らの体を突き抜けはするが、彼らにできた傷はすぐにふさがってしまう。地球人の武器は役に立たないのか。そんなある日、何とあのキンチョルンに彼らを抹殺する力があることがわかった。地球防衛軍はキンチョルンの性能を数10倍アップさせたキンチョルンIIを開発し、あなたに、侵略生物を抹殺することを命令した。さあキンチョルンを持って出動だ!!

## ジョイスティック版とキー版

もし、あなたがFM-8のオーナーで、スヒタル産業のジョイスティック(製品名、JOY-8)か富士音響マイコンセンターRAMのジョイスティックを、お持ちでしたら、ソース・プログラムとデータ・リストを入れてください。またジョイスティックを持ってないFM-8のオーナーかFM-7のオーナーの方はソース・プログラムをキー版の変更リストにしたがって入れてください。プログラムのセーブは **SAVEM "K INCHO"**、&H3800、&H617 F、&H3800で行ってください。少し長めですが、DISK BASIC下で使うようにするためにこのアドレスになってしまいました。スタート番地は&H5000です。

キンチョルン噴霧



## ルール

キンチョマンは3人、インベダーは1面4体、2面5体、3面6体、以下これの繰り返しです。ジョイスティックおよびキーの操作は図1に示すとおりです。あなたはインベダーを避けながら4箇所までわなを仕掛けられます。わなは時間がたつと自然に消えます。

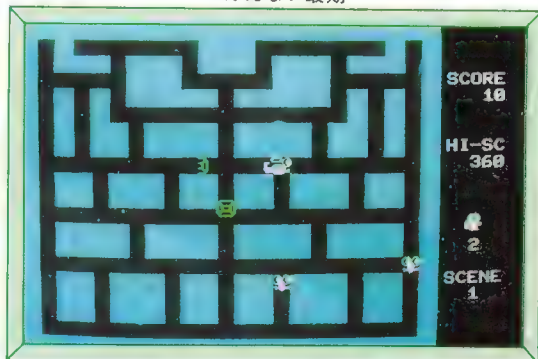
キンチョルン・ガスはある程度の時間ふきかけないとききめがありません。したがってわなにかかったインベダーのみに対して有効です。決して自由歩行しているインベダーと張り合わないでください。負けるだけです。その面でのインベダーを全部抹殺すればその面は終了します。面数が多くなるとわなの有効時間が短くなっていくのでそれだけ難しくなっていきます。

ゲームのコツは曲がり角の中心にわなをしかけ、あまり広い範囲にわなを分散しないことです。わなにかかるのはインベダーだけではありません。キンチョマンも自分で仕掛けたわなにははまりと身動きできなくなってしまいます。そうなれば道は2つに1つ、インベダーに食われるかわなが解けるまでもがくかです。作者はゲームを作るわりにするのは下手で2,000点ぐらいしかいきません。

## オマケ

ジョイスティックをすでに持っている人にも、またこれか

あえない最期





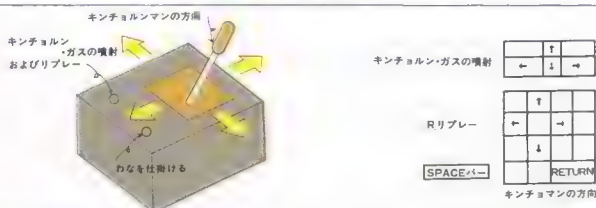
ら買う人、作る人にも役に立つおまけを付けます。それは“ザ・キューティー”をジョイスティック版に改良するためのリストです。リアルタイム・ゲームの苦手なFM-8ですが、ジョイスティックさえあれば何人か！私はジョイスティックの操作の良さを一人でも多くの人に知ってもらい、楽しんで欲しいと思います。次回発表のゲームもジョイスティック

クを使ったものを予定しています。期待してください。

#### 参考文献

- 1) "拡張Kコンパイラ" 工学社 FM-8活用研究
- 2) "バックマン"
- 3) "キューティー", I/O, '83年1月号

1



## コンパイラ・ソース・リスト

```

10 ' (***** INITIALIZE *****)
20 ' CONST DATA=$3800,WDRK=$2000,EADR=$3F60;RO=DATA;GADR=EADR+180;WADR=GADR+1
80#4;TRAD=WADR+180#4;DEAD=TRAD+90;SND=$4780
30 ' WKE=WORK;ELC=WKE+1839;ELY=ELX+10;ETO=ELY+10;EDA=ETO+10;EDD=EDA+10;TRX=ED
D+4;TRY=TRX+4;HISC=0;?CHR$(12)
40 ' INIT;T=24;G=10;ELN=2;DX=0;DY=0;DTO=0;DLP=0;DLF=0;GX=0;GY=0;GLP=0;TRN=0;X0=0
50 ' Y0=0;X1=0;Y1=0;X2=0;Y2=0;X3=0;Y3=0;MM=0;KK=0;SCRE=0;GAM=0;EXX=0
60 ' TRLH=100;MDIG=3
70 ' (***** SCREEN INITIALIZE *****)
80 ' DADR=WADR+180;DPUT(49,23,0)
90 ' LINE(0,0,551,199,0,5)
100 ' CL=7;LOC(35,3);?"SCORE";LOC(35,7);?"HI-SC";LOC(37,13);?"MDIG";LOC(35,15);
?"SCENE"
110 ' (***** WORK AREA INITIALIZE *****)
120 ' FOR I=0 TO 9;ELX:I)=0;ELY:I)=0;ELP:I)=0;ETO:I)=0;EDA:I)=0;NEXT
130 ' ADDR=$6A00;CPOKE ADDR,$00,$00,$00,$48,$00
140 ' ADDR=$6A08;CPOKE ADDR,$00,$00,$01,$48,$00
150 ' ADDR=$6A10;CPOKE ADDR,$00,$00,$02,$48,$00
160 ' ADDR=$6A18;CPOKE ADDR,$00,$00,$03,$48,$00
170 ' (***** MAIN LOOP *****)
180 ' START;GAM=GAM+1;ELN=ELN+1;IF ELN>5 THEN ELN=3;TRLH=TRLH-5;IF TRLH<50 THEN T
RLH=100;F1;F1;LOC(35,16);?#3,GAM
190 ' START1;IF MDIG=0 THEN LOC(35,14);?"GAME OVER PUSH RED BUTTON"
200 ' IF SCRE>HISC THEN HISC=SCRE;F1;LOC(35,4);?" " ;LOC(35,8);?#4,HISC,"0"
"
210 ' START2;CODE $34,$18,$86,$FD,$1F,$8B,$8E,$6A,$10,$BD,$F3,$3C,$35,$18;Q=PEEK(
$6A14);IF Q=0 THEN GOTO INIT;ELN=ELN+1;ETO=ETO+10;EDD=EDA+10;EDD:LP)=Q
220 ' FOR I=0 TO 47;FOR J=0 TO 36;DELN=ELN
230 ' Q=RO:I+J*48);IF Q THEN BOX1I,I,J;FI
240 ' NEXT;NEXT
250 ' FOR I=0 TO 1839;WKE:I)=0;NEXT
260 ' FOR I=0 TO 9;ELX:I)=22;ELY:I)=36;EDA:I)=0;EDD:I)=10;NEXT
270 ' DX=22;DY=6;DTO=1;DADR=GADR+180;DPUT(DX,DY,MM);KK=0
280 ' FOR I=0 TO 3;TRX:I)=0;TRY:I)=0;NEXT
290 ' LOOP;FOR LP=0 TO ELN
300 ' X=ELX:LP);Y=ELY:LP);XX=X;YY=Y
310 ' Q=EDD:LP);Q=Q+1;IF Q=10 THEN Q=10;F1;EDD:LP)=Q
320 ' Q=EDA:LP);IF Q THEN GOTO EXT;FI
330 ' Q=WKE:X+Y*48);IF Q THEN GOTO LOOP3;FI
340 ' ELF=RO:X+Y*48);Q=ELF
350 ' IF Q=1 THEN GOTO LOOP2;FI
360 ' LOOP1;ETO:LP)=RND(4)
370 ' LOOP2;QQ=ETO:LP);XX=X;YY=Y
380 ' IF QQ=0 THEN YY=YY-1
390 ' ELSE IF QQ=1 THEN YY=YY+1
400 ' ELSE IF QQ=2 THEN XX=XX-1
410 ' ELSE IF QQ=3 THEN XX=XX+1
420 ' FI;FI;FI;FI
430 ' ELF=RO:XX+YY*48);Q=ELF
440 ' IF Q=0 THEN GOTO LOOP1;FI
450 ' ETO:LP)=QQ
460 ' LOOP3;DADR=$3760
470 ' DADR=EADR+90;ELP
480 ' ELX:LP)=XX;ELY:LP)=YY
490 ' BOX1(X,Y);DPUT(XX,YY,MM);MM=1
500 ' IF GLP=1 THEN IF XX=GX AND YY=GY THEN Q=EDD:LP);Q=Q-2;EDD:LP)=Q;FI;FI
510 ' IF Q<=0 THEN EDA:LP)=1;ELX:LP)=0;ELY:LP)=0;SOUND(17);SCRE=SCRE+GAM;LOC(
35,4);?#4,SCRE,"0";DELN=DELN-1;IF DELN<0 THEN MM=0;GOTO START;FI;FI
520 ' EXT;NXT;MM=0;IF ELP=0 THEN ELP=1;ELSE ELP=0;FI
530 ' DIG;X=DX;Y=DY;XX=DX;YY=DY
540 ' IF DDA THEN GOTO DEXT;FI
550 ' CODE $34,$18,$86,$FD,$1F,$8B,$8E,$6A,$00,$BD,$F3,$3C,$8E,$6A,$08,$BD
560 ' CODE $F3,$3C,$8E,$6A,$10,$BD,$F3,$3C,$8E,$6A,$18,$BD,$F3,$3C,$35,$18
570 ' KY1=PEEK($6A04);KY2=PEEK($6A0C);KY=PEEK($6A14);IF KY=0 THEN GOTO GUG;FI
580 ' IF KY1<50 THEN KY=0;YY=YY-1
590 ' ELSE IF KY1>200 THEN KY=1;YY=YY+1
600 ' ELSE IF KY2<50 THEN KY=2;XX=XX-1
610 ' ELSE IF KY2>200 THEN KY=3;XX=XX+1
620 ' ELSE GOTO DEXT
630 ' FI;FI;FI;FI
640 ' DLF=RO:XX+YY*48)
650 ' IF DLF=0 THEN GOTO DEXT;FI
660 ' IF DTO<>KY THEN DTO=KY;XX=DX;YY=DY;ELSE DTO=KY;FI
670 ' DADR=GADR+180*DTO
680 ' IF DLP=0 THEN DLP=1;DADR=DADR+90;ELSE DLP=0;FI

```

## コンパイラ・ソース・リスト

```

690 ' I=0;WHILE I<=3;X1=TRX:I;Y1=TRY:I;IF X1=DX AND Y1=DY THEN XX=X;YY=Y;KK
=1;FI;I=I+1;WEND
700 ' BOX1(X,Y);DPUT(XX,YY,MM)
710 ' DX=XX;DY=YY
720 ' DEXT;JUDGE(DX,DY,ELN)
730 ' GUS;KY=PEEK(%6A14)
740 ' IF KY THEN GOTO GEXT;FI
750 ' XX=DX;YY=DY
760 ' IF DTO=0 THEN YY=YY-2
770 ' ELSE IF DTO=1 THEN YY=YY+2
780 ' ELSE IF DTO=2 THEN XX=XX-2
790 ' ELSE IF DTO=3 THEN XX=XX+2
800 ' FI;FI;FI;FI
810 ' Q=RO:XX+YY*48)
820 ' IF Q=0 THEN GOTO GEXT;FI
830 ' DADR=WADR+180*DTO;DPUT(DX,DY,MM)
840 ' DADR=DADR+90;DPUT(XX,YY,MM);SOUND[34]
850 ' GX=XX;GY=YY;GLP=1;GOTO TRAP
860 ' GEXT;IF GLP=0 THEN GOTO TRAP;FI
870 ' BOX1(GX,GY);GLP=0
880 ' TRAP;KY=PEEK(%6A1C)
890 ' IF KY OR KK THEN GOTO TEXT;FI
900 ' FOR I=0 TO 3
910 ' X=TRX:I;Y=TRY:I;Q=WKE:X+Y*48)
920 ' IF Q=0 THEN TRN=I;SW=1;FI
930 ' NEXT
940 ' IF SW=0 THEN GOTO TEXT;FI
950 ' XX=DX;YY=DY
960 ' IF DTO=0 THEN YY=YY-2
970 ' ELSE IF DTO=1 THEN YY=YY+2
980 ' ELSE IF DTO=2 THEN XX=XX-2
990 ' ELSE IF DTO=3 THEN XX=XX+2
1000 ' FI;FI;FI;FI
1010 ' Q=RO:XX+YY*48)
1020 ' IF Q=0 THEN SW=0;GOTO TEXT;FI
1030 ' I=0;WHILE I<=3;X=TRX:I;Y=TRY:I;IF XX=X AND YY=Y THEN GOTO TEXT;FI;I=
I+1;WEND
1040 ' DADR=TRAD
1050 ' DPUT(XX,YY,MM);TRX:TRN)=XX;TRY:TRN)=YY
1060 ' WKE:XX+YY*48)=TRLH;SW=0
1070 ' TEXT;
1080 ' FOR I=0 TO 3
1090 ' XX=TRX:I;YY=TRY:I)
1100 ' Q=WKE:XX+YY*48);IF Q=0 THEN GOTO TNEX;FI
1110 ' Q=Q-1;WKE:XX+YY*48)=Q
1120 ' IF Q=0 THEN BOX1(XX,YY);TRX:I)=0;TRY:I)=0;KK=0;ELSE DADR=TRAD;TPUT(XX,
YY,MM);MM=1;FI
1130 ' TNEX;NEXT;MM=0
1140 ' TOUT;JUDGE(DX,DY,ELN);GOTO LOOP
1150 ' (***** SUBROUTINES *****)
1160 ' JUDGE;X0=%1 ;Y0=%2-1
1170 ' X1=%1 ;Y1=%2+1
1180 ' X2=%1-1;Y2=%2
1190 ' X3=%1+1;Y3=%2
1200 ' FOR LP=0 TO %3
1210 ' XX=ELX:LP;YY=ELY:LP)
1220 ' IF %1=XX AND %2=YY THEN GOTO DETH;FI
1230 ' IF X0=XX AND Y0=YY THEN GOTO DETH;FI
1240 ' IF X1=XX AND Y1=YY THEN GOTO DETH;FI
1250 ' IF X2=XX AND Y2=YY THEN GOTO DETH;FI
1260 ' IF X3=XX AND Y3=YY THEN GOTO DETH;FI
1270 ' NEXT
1280 ' RETURN
1290 ' DETH;SOUND[0];DADR=DEAD;MM=0;MDIG=MDIG-1;LOC[37,13];?MDIG
1300 ' BOX1(DX,DY);DPUT(DX,DY,MM)
1310 ' FOR I=0 TO 21000;NEXT
1320 ' GOTO START1
1330 ' SOUND;SD=%1;LO=SND:SD;SD=SD+1
1340 ' FOR J=0 TO LO
1350 ' L1=SND:SD;L2=SND:SD+1)
1360 ' FOR I=0 TO L1
1370 ' FOR WA=0 TO L2;NEXT
1380 ' POKE$FD03,$B1
1390 ' FOR WA=0 TO L2;NEXT
1400 ' POKE$FD03,1
1410 ' NEXT;SD=SD+2
1420 ' NEXT;RETURN
1430 ' (* GRAPHIC SUBROUTINES *)
1440 ' DPUT;X1=%1*12;Y1=%2*5;PUT(X1,Y1,X1+23,Y1+9,DADR,90,%3,0);RETURN
1450 ' TPUT;X1=%1*12;Y1=%2*5;PUT(X1,Y1,X1+23,Y1+9,DADR,90,%3,2);RETURN
1460 ' BOX1;X1=%1*12;Y1=%2*5;LINE(X1,Y1,X1+T-1,Y1+G-1,0,0);RETURN
1470 ' PUT;IF %7 THEN HLT[1];ADDR=$FCB2;CPOKE ADDR,$1E,%1*,%2*,%3*,%4*,0,%B,90;SUB
[1];RETURN;FI
1480 ' QE=%6-1;HLT[1];ADDR=$FCB2;CPOKE ADDR,$1E,%1*,%2*,%3*,%4*,0,%B,90;DTD=%5;
FOR WKI=0 TO QE;ADDR=WKI)=DTD:WKI);NEXT;SUB[1];RETURN
1490 ' LINE;HLT[1];ADDR=$FCB2;CPOKE ADDR,$15,%6,%5,%1*,%2*,%3*,%4*;SUB[1];RETURN
1500 ' LOC;HLT[1];ADDR=$FCB2;CPOKE ADDR,3,5,%12,%1,%2,%11,CL+%80;SUB[1];RETURN
1510 ' CON;HLT[1];ADDR=$FCB2;CPOKE ADDR,1,0,40,25,0,25,0,1,0;SUB[1];RETURN
1520 ' HLT;CODE $B6,$FD,$05,$2B,$FB;POKE $FD05,$B0;CODE $B6,$FD,$05,$2A,$FB;RETUR
N
1530 ' SUB;POKE $FD05,0;RETURN

```

## データ・ダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3800	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
3810	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
3820	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
3830	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
3840	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
3850	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
3860	00	00	02	00	00	00	02	01	01	02	01	01	01	01	01	01	0E
3870	01	01	02	00	00	00	00	00	00	02	01	01	01	01	01	01	0B

3880	01	01	02	01	01	01	02	00	00	00	02	00	00	00	00	00	0B
3890	00	00	01	00	00	00	00	00	00	00	00	01	00	00	00	00	02
38A0	00	00	01	00	00	00	00	00	00	00	00	01	00	00	00	00	02
38B0	00	00	01	00	00	00	00	00	00	00	00	01	00	00	00	00	02
38C0	00	00	01	00	00	00	00	00	00	00	00	01	00	00	00	00	02
38D0	00	00	01	00	00	00	00	00	00	00	00	01	00	00	00	00	02
38E0	00	00	01	00	00	00	00	00	00	00	00	01	00	00	00	00	02
38F0	00	00	01	00	00	00	00	00	00	00	00	01	00	00	00	00	02

Sums	02	02	0D	01	01	01	04	01	01	01	0D	02	02	03	02	02	+32
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3900	00	00	01	00	00	00	00	00	00	00	00	01	00	00	00	00	00
3910	00	00	01	00	00	00	00	00	00	00	00	01	00	00	00	00	00
3920	00	00	02	01	01	01	02	01	01	01	02	00	00	00	00	00	00
3930	00	00	02	01	01	01	02	01	01	01	02	00	00	00	00	00	00
3940	00	00	02	01	01	01	02	01	01	01	02	00	00	00	00	00	00
3950	00	00	01	00	00	00	01	00	00	00	01	00	00	00	00	00	00
3960	00	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	01
3970	00	00	01	00	00	00	01	00	00	00	01	00	00	00	00	00	00
3980	00	00	01	00	00	00	01	00	00	00	01	00	00	00	00	00	00
3990	00	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
39A0	00	00	01	00	00	00	01	00	00	00	01	00	00	00	00	00	00
39B0	00	00	01	00	00	00	01	00	00	00	01	00	00	00	00	00	00
39C0	00	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
39D0	00	00	01	00	00	00	01	00	00	00	01	00	00	00	00	00	00
39E0	00	00	01	00	00	00	01	00	00	00	02	01	02	01	01	01	00
39F0	01	01	01	01	01	01	02	01	01	01	01	01	01	01	01	01	01
Sums	01	01	10	04	04	04	12	04	04	04	11	02	03	02	02	02	+5B
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3A00	02	01	02	00	00	00	01	00	00	00	+A	+B	+C	+D	+E	+F	00
3A10	00	00	01	00	00	00	01	00	00	00	00	00	01	00	00	00	00
3A20	00	00	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00
3A30	01	00	00	00	00	00	00	01	00	00	00	01	00	00	00	00	00
3A40	00	00	01	00	00	00	01	00	00	00	00	00	01	00	00	00	00
3A50	00	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
3A60	01	00	00	00	00	00	01	00	00	00	01	00	00	00	00	00	00
3A70	00	00	01	00	00	00	01	00	00	00	00	00	01	00	00	00	00
3A80	00	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
3A90	01	00	00	00	00	00	01	00	00	00	01	00	00	00	00	00	00
3AA0	00	00	01	00	00	00	01	00	00	00	00	00	01	00	00	00	00
3AB0	00	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
3AC0	01	00	00	00	00	00	01	00	00	00	01	00	00	00	00	00	00
3AD0	00	00	01	00	00	00	01	00	00	00	00	00	01	00	00	00	00
3AE0	00	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
3AF0	01	00	00	00	00	00	01	00	00	00	01	00	00	00	00	00	00
Sums	07	01	07	00	00	00	10	00	00	00	06	00	05	00	00	00	+2A
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3B00	00	00	02	01	01	01	02	01	02	01	+A	+B	+C	+D	+E	+F	00
3B10	02	01	01	01	01	01	02	01	01	01	01	01	02	01	01	01	00
3B20	02	01	01	01	02	01	02	01	01	01	02	00	00	00	00	00	00
3B30	00	00	01	00	00	00	00	00	01	00	00	00	00	00	00	00	00
3B40	01	00	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00
3B50	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00	00	00
3B60	00	00	01	00	00	00	00	01	00	00	00	00	00	00	00	00	00
3B70	01	00	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00
3B80	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00	00	00
3B90	00	00	01	00	00	00	00	01	00	00	00	00	00	00	00	00	00
3BA0	01	00	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00
3BB0	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00	00	00
3BC0	00	00	01	00	00	00	00	01	00	00	00	00	00	00	00	00	00
3BD0	01	00	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00
3BE0	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00	00	00
3BF0	00	00	01	00	00	00	00	01	00	00	00	00	00	00	00	00	00
Sums	0B	02	0B	03	0B	03	0A	03	09	03	0B	02	0B	02	02	02	+52
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3C00	01	00	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00
3C10	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00	00	00
3C20	00	00	02	01	01	01	01	02	01	01	01	02	01	01	01	01	00
3C30	02	01	01	01	01	01	02	01	01	01	01	02	02	01	01	01	00
3C40	02	01	01	01	02	01	01	01	01	01	02	00	00	00	00	00	00
3C50	00	00	01	00	00	00	00	00	00	00	00	00	01	00	00	00	00
3C60	00	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
3C70	01	00	00	00	00	00	00	00	00	00	01	00	00	00	00	00	00
3C80	00	00	01	00	00	00	00	00	00	00	00	00	01	00	00	00	00
3C90	00	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
3CA0	01	00	00	00	00	00	00	00	00	00	01	00	00	00	00	00	00
3CB0	00	00	01	00	00	00	00	00	00	00	00	00	01	00	00	00	00
3CC0	00	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
3CD0	01	00	00	00	00	00	00	00	00	00	01	00	00	00	00	00	00
3CE0	00	00	01	00	00	00	00	00	00	00	00	00	01	00	00	00	00
3CF0	00	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
Sums	09	02	0B	03	07	03	0B	03	07	03	09	02	0B	02	02	02	+4E
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3D00	01	00	00	00	00	00	00	00	00	00	01	00	00	00	00	00	00
3D10	01	00	00	00	00	00	00	00	00	00	00	00	00	01	00	00	00
3D20	00	00	00	00	00	00	01	00	00	00	00	00	00	00	00	00	00
3D30	01	00	00	00	00	00	00	00	00	00	01	00	00	00	00	00	00
3D40	00	00	02	01	01	01	01	01	02	01	01	02	01	02	01	01	00
3D50	02	01	01	01	01	01	02	01	01	01	01	02	01	02	01	01	00
3D60	02	01	01	01	02	01	01	01	01	01	02	00	00	00	00	00	00
3D70	00	00	01	00	00	00	00	01	00	00	00	00	00	00	00	00	00
3D80	01	00	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00
3D90	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00	00	00
3DA0	00	00	01	00	00	00	00	01	00	00	00	00	00	00	00	00	00
3DB0	01	00	00	00	00	00	01	00	00	00	00	00	00	01	00	00	00
3DC0	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00	00	00
3DD0	00	00	01	00	00	00	00	00	01	00	00	00	00	00	00	00	00
3DE0	01	00	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00
3DF0	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00	00	00
Sums	09	02	0B	03	07	03	0B	03	07	03	09	02	0B	02	02	02	+4E
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3E00	00	00	01	00	00	00	00	00	01	00	00	00	00	00	00	00	00
3E10	01	00	00	00	00	00	00	01	00	00	00	00	00	01	00	00	00
3E20	00	00	00	00	01	00	00	00	00	00	00	01	00	00	00	00	00

3E30	00	00	01	00	00	00	00	00	01	00	00	00	00	00	00	00	:02
3E40	01	00	01	00	00	00	01	00	00	00	00	01	00	00	00	00	:03
3E50	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00	00	:02
3E60	00	00	01	00	00	00	00	00	01	00	00	00	00	00	00	00	:02
3E70	01	00	00	00	00	00	01	00	00	00	00	00	01	00	00	00	:03
3E80	00	00	00	00	01	00	00	00	00	01	00	00	00	00	00	00	:02
3E90	00	00	01	00	00	00	00	00	01	00	00	00	00	00	00	00	:02
3EA0	01	00	00	00	00	00	01	00	00	00	00	00	01	00	00	00	:03
3EB0	00	00	00	00	01	00	00	00	00	00	01	00	00	00	00	00	:02
3EC0	00	00	02	01	01	01	01	01	02	01	01	01	01	01	01	01	:10
3ED0	02	01	01	01	01	01	02	01	01	01	01	02	01	01	01	01	:13
3EE0	01	01	01	01	02	01	01	01	01	01	02	00	00	00	00	00	:0B
3EF0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
Sum:	07	02	0B	03	0B	03	0B	03	0B	03	0B	02	07	02	02	02	:4C
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3F00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3F10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3F20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3F30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3F40	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3F50	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
3F60	00	00	00	1B	01	B0	7E	07	0F	7E	07	0E	1B	19	B0	00	:14
3F70	7E	00	00	FF	00	01	FF	B0	1F	FF	B8	3B	3C	1C	00	00	:A3
3F80	00	1B	01	B0	7E	07	0F	7E	07	0F	0E	1B	19	B0	00	00	:92
3F90	00	FF	00	01	FF	B0	1F	FF	FF	B8	3B	3C	1C	1F	B1	B8	:7B
3FA0	67	B6	FE	1F	1E	FE	1F	1E	7B	67	86	1F	B1	FB	00	00	:E6
3FB0	00	00	00	00	00	00	00	00	00	00	00	00	00	01	B0	1B	:99
3FC0	07	0E	7E	07	0E	7E	01	9B	18	00	7E	00	00	FF	00	39	:33
3FD0	FF	9C	1F	FF	FB	00	3C	00	00	00	00	01	B0	1B	07	0E	:6D
3FE0	7E	07	0E	7E	01	9B	18	00	7E	00	00	FF	00	39	FF	9C	:E5
3FF0	1F	FF	FB	00	3C	00	00	1F	B1	FB	61	E6	1E	B7	FB	7F	:0A
Sum:	8B	1F	74	3B	74	1C	0F	FE	04	5D	0F	3D	BA	7B	F7	FB	:54
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4000	F8	7F	61	E6	1E	1F	B1	F8	00	00	00	00	00	00	00	00	:74
4010	00	00	00	00	01	FF	B0	07	FF	0F	0F	FF	F0	07	FF	0F	:4B
4020	01	FF	B0	1F	FF	C0	1C	FF	C0	00	FF	00	00	E0	00	00	:1B
4030	00	00	01	FF	B0	07	FF	0F	0F	0F	0F	07	FF	E0	01	FF	:4A
4040	B0	00	FF	00	00	FF	00	00	FF	00	00	E0	E0	B1	E7	B0	:1C
4050	01	FF	B0	07	FF	E0	0F	FF	0F	07	FF	E0	01	FF	B0	1F	:E9
4060	FF	C0	1C	FF	C0	00	FF	00	E0	E7	B0	01	E7	B0	01	FF	:68
4070	B0	07	FF	E0	0F	FF	F0	07	FF	E0	01	FF	B0	03	FF	FB	:A4
4080	03	FF	3B	00	FF	00	00	07	00	00	00	00	01	FF	B0	07	:C7
4090	FF	E0	00	FF	00	F0	07	FF	E0	01	FF	B0	00	FF	00	00	:F4
40A0	00	00	FF	00	01	E7	00	01	E7	B0	01	FF	B0	07	FF	E0	:B5
40B0	0F	FF	F0	07	FF	E0	01	FF	B0	03	FF	FB	03	FF	3B	00	:98
40C0	FF	00	01	E7	00	01	E7	B0	01	FF	B0	07	FF	E0	0F	1B	:DC
40D0	F0	07	FF	E0	01	FF	B0	03	FF	FB	03	FF	3B	00	FF	00	:89
40E0	00	00	00	00	00	00	01	FF	B0	06	06	00	00	00	30	06	:07
40F0	00	60	01	FF	B0	00	01	7E	00	0F	00	00	FF	00	00	00	:53
Sum:	F9	69	B3	B7	DC	91	00	4D	A4	3B	B1	2A	9C	2F	5C	60	:97
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4100	B0	01	E7	B0	03	FF	F8	03	FF	0F	0F	00	00	E7	B0	01	:1B
4110	01	FF	B0	03	FF	F8	03	FF	0F	0F	0F	00	00	E7	B0	01	:1B
4120	E7	B0	01	FF	B0	07	FF	E0	0F	1B	F0	07	FF	E0	01	FF	:3A
4130	B0	1F	FF	C0	1C	FF	E0	00	FF	00	00	07	00	00	00	00	:0F
4140	01	FF	B0	06	00	60	0C	00	30	06	00	60	01	FF	B0	00	:0B
4150	7E	00	00	F0	00	00	FF	00	01	E7	00	01	E7	B0	01	FF	:08
4160	B0	07	FF	E0	0F	1B	F0	07	FF	E0	01	FF	B0	1F	FF	C0	:C1
4170	1C	FF	C0	00	FF	00	01	E7	00	01	E7	B0	01	FF	B0	07	:1B
4180	FF	E0	0C	67	F0	07	FF	E0	01	FF	B0	00	FF	B0	00	FF	:26
4190	B0	01	FF	C0	01	C1	C0	00	00	00	00	01	FF	B0	00	01	:E0
41A0	00	00	F0	00	01	E0	01	FF	B0	00	FC	00	00	E0	00	01	:2E
41B0	FF	C0	01	C1	C0	07	C7	C0	01	FF	B0	01	FF	B0	07	FF	:67
41C0	F0	07	FF	E0	01	FF	B0	00	FF	B0	00	FF	B0	01	FF	C0	:14
41D0	01	C1	C0	07	C7	C0	01	FF	B0	07	FF	E0	0C	63	F0	07	:DC
41E0	FF	E0	01	FF	B0	00	FF	F0	07	FF	70	00	FF	00	00	77	:3A
41F0	00	00	00	00	00	01	FF	B0	00	01	E0	00	00	F0	00	01	:02
Sum:	71	ED	62	E6	A5	E2	C5	62	7E	2A	52	EB	51	0F	7D	0B	:21
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4200	01	FF	B0	01	FF	B0	07	FF	E0	0F	0F	00	00	00	77	00	:F8
4210	BE	00	FF	07	00	FF	1F	00	FF	0F	1F	00	E7	1F	01	E7	:80
4220	B0	00	FF	F0	0F	FF	0F	0F	0F	07	FF	E0	01	FF	B0	01	:3D
4230	01	FF	B0	07	FF	E0	0F	C6	30	07	FF	E0	01	FF	B0	01	:0A
4240	FF	00	01	FF	00	33	FF	B0	03	B3	B0	00	00	00	01	FF	:87
4250	B0	07	B0	00	0F	00	00	07	B0	00	01	FF	B0	00	3F	00	:5C
4260	00	00	07	03	FF	B0	03	B3	B0	03	E3	E0	01	FF	B0	07	:DC
4270	FF	E0	0F	C6	30	07	FF	E0	03	E3	E0	01	FF	B0	01	FF	:4A
4280	00	03	FF	B0	03	B3	B0	03	E3	E0	01	FF	B0	07	FF	E0	:4C
4290	0F	C6	30	07	FF	E0	01	FF	B0	0F	0F	00	0E	FF	E0	00	:66
42A0	FF	00	00	EE	00	00	00	00	01	FF	B0	07	B0	00	0F	00	:03
42B0	00	07	B0	00	01	FF	B0	00	3F	00	00	00	FF	00	00	FF	:44
42C0	00	EE	00	00	FB	00	01	FF	B0	07	FF	E0	0F	C6	30	07	:5B
42D0	FF	E0	01	FF	B0	0F	FF	00	0E	FF	E0	00	FF	00	00	EE	:47
42E0	00	00	FB	00	01	FF	B0	07	FF	C0	0F	FF	F0	07	FF	E0	:25
42F0	01	FF	B0	1F	FF	FB	1C	FF	00	00	FF	00	00	E7	00	00	:97
Sum:	ED	B9	BB	51	3E	D8	1C	96	6F	A3	3F	22	8C	0F	3D	BA	:4F
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4300	00	00	01	FF	B0	00	FF	E0	0F	0F	00	00	00	00	01	FF	:2A
4310	BE	00	FF	07	00	FF	1F	00	FF	0F	1F	00	E7	1F	01	E7	:80
4320	01	FF	B0	07	FF	C0	0F	0F	0F	07	FF	E0	01	FF	B0	1F	:09
4330	FF	FF	1C	FF	1F	00	FF	1F	00	E7	1F	01	E7	B0	33	33	:2A
4340	33	CC	CC	CC	03	33	33	00	CC	CC	00	33	33	00	CC	CC	:16
4350	00	CC	03	33	00	CC	00	00	33	00	00	CC	33	33	00	CC	:E6
4360	CC	CC	03	33	03	33	00	CC	CC	00	33	33	00	CC	CC	00	:16



## データ・ダンプ・リスト

4370	33	00	00	CC	00	00	33	00	00	0C	33	33	33	0C	CC	CC	:78
4380	03	33	33	00	CC	CC	00	33	33	00	0C	CC	00	03	33	00	:75
4390	00	CC	00	00	33	00	00	0C	01	FF	80	07	FF	E0	0F	18	:98A
43A0	F0	07	FF	E0	01	FF	80	07	FF	F8	00	FF	38	00	FF	00	:8A
43B0	00	E7	00	00	00	00	01	FF	80	06	00	60	0C	00	30	06	:0F
43C0	00	60	71	FF	B0	F8	3E	00	F8	FF	00	F8	FF	00	F8	E7	:53
43D0	00	01	E7	80	01	FF	80	07	FF	E0	0F	18	F0	07	FF	E0	:CB
43E0	01	FF	B0	FF	FF	F8	FF	F8	FF	F8	FF	00	F8	E7	00	01	:7C
43F0	E7	B0	30	00	00	CC	00	00	33	00	00	CC	C0	00	33	30	:B5
-----																	
Sum:	9B	A6	D8	35	54	48	95	F5	12	EB	0E	4F	95	3C	41	8E	:71
-----																	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4400	00	CC	CC	00	33	33	00	CC	CC	C0	33	33	30	CC	CC	CC	:50
4410	30	00	00	CC	00	00	33	00	00	CC	C0	00	33	30	00	CC	:EA
4420	CC	00	33	33	00	CC	CC	00	33	33	30	CC	CC	CC	30	00	:84
4430	00	CC	00	00	33	00	00	CC	C0	00	33	30	00	CC	CC	00	:B6
4440	33	33	00	CC	CC	00	33	33	30	CC	CC	CC	00	FF	C0	03	:7A
4450	FF	F0	06	31	F8	03	FF	F0	00	FF	C0	00	7F	C0	07	FF	:14
4460	C0	00	FF	E0	00	E0	E0	00	00	00	00	FF	C0	00	00	F0	:0E
4470	00	00	7B	00	00	F0	70	FF	C0	F8	7E	00	F8	70	00	F8	:6D
4480	FF	E0	F8	E0	E0	E3	E0	00	FF	C0	03	FF	F0	06	31	:A5	
4490	F8	03	FF	F0	00	FF	C0	F8	7F	C0	FF	FF	C0	F8	FF	E0	:75
44A0	F8	E0	E0	E3	E0	30	00	00	CC	C0	00	33	33	00	CC	:6C	
44B0	CC	CC	33	33	33	CC	CC	CC	33	33	00	CC	C0	00	30	00	:B7
44C0	00	00	00	30	00	00	00	CC	C0	00	33	33	00	CC	CC	CC	:86
44D0	33	33	33	CC	CC	CC	33	33	00	CC	C0	00	30	00	00	:1F	
44E0	00	00	30	00	00	CC	C0	00	33	33	00	CC	CC	CC	33	:CC	
44F0	33	CC	CC	CC	33	33	00	CC	C0	00	30	00	00	00	00	:B9	
-----																	
Sum:	0F	49	B5	7A	4F	0B	13	E9	14	3F	02	C7	14	76	C3	5E	:A4
-----																	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4500	03	FF	00	0F	FF	C0	1F	BC	60	0F	FF	C0	03	FF	00	03	:AE
4510	FE	00	03	FF	E0	07	FF	00	07	07	00	00	00	00	03	FF	:E9
4520	00	0F	00	00	1E	00	00	0F	00	00	03	FF	0E	00	7E	1F	:F6
4530	00	07	1F	07	FF	1F	07	07	1F	07	C7	C0	03	FF	00	0F	:17
4540	FF	C0	1F	8C	60	0F	FF	C0	03	FF	00	03	FE	1F	03	FF	:8C
4550	FF	07	FF	1F	07	07	1F	07	C7	C0	00	00	00	00	03	33	:21
4560	00	CC	CC	33	33	33	CC	CC	CC	33	33	33	00	CC	CC	00	:C6
4570	03	33	00	00	00	00	00	00	00	00	00	00	03	33	00	CC	:50
4580	CC	33	33	33	CC	CC	CC	33	33	33	00	CC	CC	00	03	33	:30
4590	00	00	00	00	00	00	00	00	00	03	33	00	CC	CC	33	:19	
45A0	33	33	CC	CC	CC	33	33	33	00	CC	CC	00	03	33	00	00	:33

45B0	0C	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:0C
45C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
45D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
45E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
45F0	0F	FF	F0	30	00	0C	CC	00	33	C3	FF	C3	C3	00	C3	C3	:07
-----																	
Sum:	1C	40	07	22	3A	3A	DA	9B	8E	D1	D6	77	B3	1B	E5	S7	:24
-----																	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4600	00	C3	C3	FF	C3	CC	00	33	30	00	0C	0F	FF	F0	00	1F	:A0
4610	FC	00	60	03	00	1F	FC	00	00	00	70	03	FC	70	0F	FF	:67
4620	7F	FF	FF	7F	FF	FF	1F	FF	FF	00	03	FC	00	1F	FC	00	:31
4630	60	03	00	1F	FC	7E	00	00	70	00	00	70	0C	03	7F	FC	:66
4640	03	7B	0C	03	1F	FF	FF	00	03	FC	00	1F	FC	00	60	03	:24
4650	00	1F	FC	7E	00	00	70	03	FC	70	0F	7F	FF	FF	7F	:82	
4660	FF	FF	1F	FF	FF	00	03	FC	00	00	00	00	00	00	00	:00	:1A
4670	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	:00
4680	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	:00
4690	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	:00
46A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	:00
46B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	:00
46C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	:00
46D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	:00
46E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	:00
46F0	00	00	00	00	5A	37	BA	00	B8	01	C7	00	AF	01	B0	:03	:2E
-----																	
Sum:	DD	5B	49	7A	13	21	8D	E9	9F	33	BE	4B	83	31	E9	9F	:BC
-----																	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
4700	30	00	30	00	40	00	37	60	38	14	3A	E4	40	00	47	2F	:57
4710	47	39	47	43	47	4D	47	57	00	18	00	0A	00	03	00	:16	:77
4720	00	24	00	02	00	00	00	01	00	00	00	16	00	24	00	:00	:61
4730	00	0A	00	25	00	01	00	07	FC	8D	00	03	00	24	00	:22	:09
4740	00	01	00	01	00	24	00	23	37	BA	00	00	00	7D	00	:7E	:34
4750	01	B0	00	AF	00	59	37	BA	00	5A	00	00	00	00	00	:00	:04
4760	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	:00
4770	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	:00
4780	08	20	40	20	38	20	30	20	28	20	20	20	18	20	10	:20	:20
4790	08	08	20	20	20	15	20	20	15	20	20	15	20	20	15	:20	:AF
47A0	20	15	01	05	05	05	05	00	00	00	00	00	00	00	00	:00	:4A
47B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00	:00
-----																	
Sum:	AB	55	D8	5F	E4	05	0A	DC	B3	02	7A	C4	78	FE	77	A6	:89

## キー変更リスト

```

130 '
140 '
150 '
160 '
190 'START1:IF MDIG=0 THEN LOC[5,14]:?"GAME OVER HIT [ R ] KEY"
210 'START2:Q=KEY[1]:IF Q=82 THEN GOTO INIT:ELSE GOTO START2:F1:F1
550 '
560 '
570 ' KY=KEY[1]
580 ' IF KY=56 THEN KY=0:YY=YY-1
590 ' ELSE IF KY=50 THEN KY=1:YY=YY+1
600 ' ELSE IF KY=52 THEN KY=2:XX=XX-1
610 ' ELSE IF KY=54 THEN KY=3:XX=XX+1
620 ' ELSE GOTO DEXT
630 ' F1:F1:F1:F1
700 ' BOX1[X,Y]:DPUT[XX,YY,MM]
730 'GUS:KY=KEY[1]
740 ' IF KY=30 THEN DTO=0:ELSE IF KY=31 THEN DTO=1:ELSE IF KY=28 THEN DTO=3:ELSE
LSE IF KY=29 THEN DTO=2:ELSE GOTO GEXT:F1:F1:F1
750 ' XX=DX:YY=DY:IF MTO>DTO THEN MTO=DTO:GOTO GEXT:F1
840 ' DADR=DADR+90:DPUT[XX,YY,MM]:SOUND[34]:MTO=DTO
880 'TRAP:KY=KEY[1]
890 ' IF KY>32 OR KK THEN GOTO TEXT:F1
1540 'KEY:HLT[1]:ADDR=$FCB2:CPOKE ADDR,$3F,'Y','A','M','A','U','C','H','I',$93,$D
3BFH,$90,$B6,$D401H,$B7,$D3B2H,$B6,$D40AH,$39:SUB[1]:HLT[1]:KY=PEEK($FCB2):SUB[1]:D
MY=KY:RETURN

```

## ザ・キューティ変更リスト

```

5 'ADDR=$4B00:CPOKE ADDR,$00,$00,$00,$4B,$00,$00,$00,$00,$00,$00,$00,$01,$4B,$00,$00
,$00,$00,$00,$00,$02,$4B,$00,$00,$00
590 'KEY:CODE $24,$1B,$B6,$FD,$1F,$B8,$BE,$4B,$00,$BD,$F3,$3C,$BE,$4B,$0B,$BD,$F
3,$3C,$BE,$4B,$10,$BD,$F3,$3C,$35,$1B
591 'F1=PEEK($4B04):F2=PEEK($4B0C):F3=PEEK($4B14):IF F1<50 THEN N=8:ELSE IF F1>2
00 THEN N=5:ELSE IF F2<50 THEN N=4:ELSE IF F2>200 THEN N=6:ELSE IF F3=0 THEN N=0
:ELSE N=2:F1:F1:F1:F1:F1:RETURN

```

## ゴルフ・シミュレーション



写真1 ティーランドから見わたす

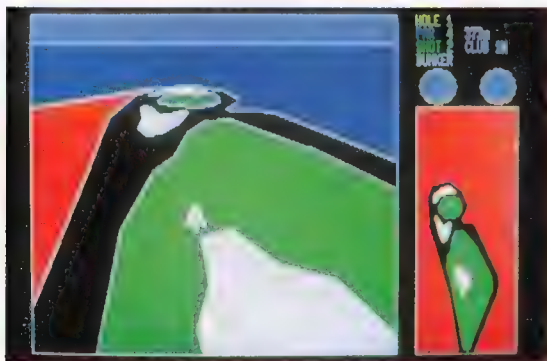
写真2 ティーショット



写真3 バンカーに落ちた!



写真4 やっとグリーン・オン



まずは、青木さん「ハワイオープン」劇的優勝おめでとうございます。

本ゴルフ・シミュレーションは、初めて3次元立体図形処理を使った画期的ゴルフ・シミュレーションです。従来のゴルフ・シミュレーションはすべて平面コースで、せいぜいグリーン回りが拡大されるぐらいでしたが、本シミュレーションでは毎回のショット位置から立体視野画面を表示し、ボールも放物線を描きながら立体的に飛んでゆきます。また、ショット方向を角度で指定するのは非常にわかりにくいので、矢印、力のバーなどを使って、できるだけ視覚的にしてあります。

目の覚める美しい海岸コース（18ホール）を充分に楽しみください。

### プログラムについて

プログラムは、メイン・プログラムと18の3次元ホール・データから成っています。

5000行からが3次元線画およびペイント・ルーチンです。立体視野画面の作成、表示およびボールの弾道などにこのルーチンを使っています。また、フェアウェイ、ラフなどの反発係数、および角度、風速、方向、クラブによる打上げ角度など、ゴルフに必要なすべての要因を計算式に組み込み、実際のゴルフをほぼ完璧にシミュレートしています。

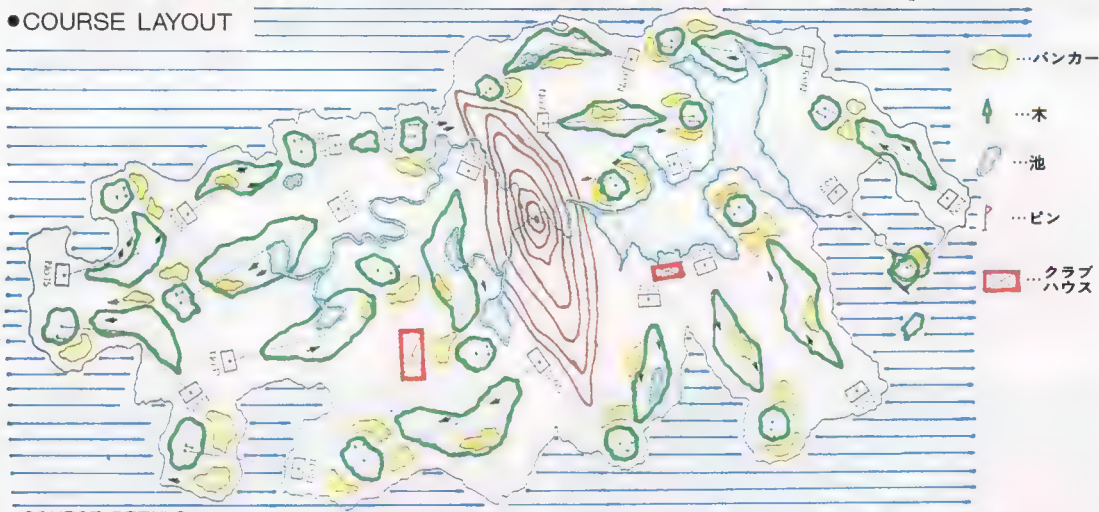
立体視野画面はショットのたびに描き変えます。当然、ショット位置は毎回違うので、ほとんど無限に近い視野画面が表示されることになります。

風はゲーム性を増すため実際より強く影響するようにし



## T&E ISLAND GOLF COURSE

### ●COURSE LAYOUT



### ●COURSE DETAILS

HOLE	1	2	3	4	5	6	7	8	9	OUT	10	11	12	13	14	15	16	17	18	IN	TOTAL
METER	362	508	172	332	358	467	382	147	348	3,076	373	155	332	441	366	372	328	138	482	2,987	6,063
PAR	4	5	3	4	4	5	4	3	4	36	4	3	4	5	4	4	4	3	5	36	72

## コースの紹介

T & Eアイランド・ゴルフ・コース はT & E開発室で独自に設計されたもので、在来ゴルフ・コースに、より戦略性をもたせ、視覚的にも個々の色を強調し、非常に美しく仕上がっています。

No 1 ホール：コースはほぼ直線で、一打めはフェアウェイ左のクロス・バンカーと右サイドの海(OB)に注意！

No 2 ホール：距離のあるロングホール。フェアウェイはうねりがきつく、2オンはまず不可能。グリーンは池に囲まれ、手前の大バンカーは相当なプレッシャーとなる。

No 3 ホール：海に面した景観のすばらしいショート・ホール。距離もあり、風の影響も受け易いので、クラブの選択は慎重に！

No 4 ホール：コース両側が海で、一打めの落下地点のフェアウェイが狭くなっている。落し場所に注意！

No 5 ホール：フェアウェイ200m地点に細長い池が横切っています。ティーショットは手前で止めるか、ギャンブルを試みるか！

No 6 ホール：大きく左にドッグレッグしているミドル・ホール。セカンド・ショットは池を気にせず大胆に！

No 7 ホール：フェアウェイ中央のヒョウタン型大クロス・バンカーをクリアすれば、好スコアが期待できる。

No 8 ホール：距離はないが、グリーンまわりはガード・バンカーでしっかり固められたやっかいなショート・ホール。

No 9 ホール：アウトの最終ホールは海に面した右ドッグレッグの美しいホール。セカンド・ショットは直接ピンを狙うと、せり出した海にドボンの危険性が高い。

No 10 ホール：フェアウェイに木が点在していて、クロス・バンカー

を避けようとするこれが障害となる。

No 11 ホール：距離155mのショート・ホール。深く切れ込んだ入江を越してのティーショット。クラブは？「アルタスなら7番ですよ」

No 12 ホール：グリーン後方のマウント・テスをめがけてティーショット。フェアウェイ左の池はナイス・ショットでも絶対に越せません。

No 13 ホール：フェアウェイはほぼ直線の雄大なロング・ホール。風向きが良ければバーディーの可能性が高い！

No 14 ホール：海岸沿いの美しいミドル・ホール。冒険は避けてフェアウェイは右サイドから攻めるのが得策。

No 15 ホール：湾に沿って左に直角にドッグレッグした、当コースの名物ホール。風向きと運に恵まれればワンオンも可能。

No 16 ホール：グリーン手前は張り出した海岸線と、池によってしっかりとガードされている。セカンド・ショットは慎重に。

No 17 ホール：当コース中、最も距離のないショート・ホール。ピンは難しい位置に切っていますが、おもいきってホールインワンを狙ってみましょう。

No 18 ホール：視野右手にクラブ・ハウスが見えてきて、いよいよ最終18番ロング・ホール。フェアウェイ左サイドの大きな池をうまくクリアすれば、あなたにも逆転のイーグル・チャンスが…。

てあります。

各ホール・データは、全コース、レイアウトに忠実に作ってありますが、データ量の関係から多少簡素化してあります。また、メモリが足りないのに、これらのデータはホールごとにカセットから読むようにしてあります。

最後に、視野画面の一部がペイントされないことがありますが、ゲーム続行には影響ないのでご了承ください。

## プログラムの入力

まず、メイン・プログラム“GS”(リスト1)を入力し、カセットにセーブします。

次にコース・データ、“OUT”(リスト2)を入力し、RUNしてください(RUNする前に別のカセットにセーブしておいた方がよいでしょう)。

RUNするとファイル・ネームを聞いてくるので“H1”を入力してください。“H1”を入力すると、ホール1のデータがカセット・レコーダに出力されます。これが終わると、またファイル・ネームを聞いてくるので、“H2”…“H9”まで同様に行なってください。“H9”まで終了すると、“Out Of Data In 30”と表示されますが、これは関係ありません。

# 3-Dゴルフ・シミュレーション

せん。

同様に“IN”(リスト3)を入力し、“H10”~“H18”までをカセットにセーブしてください。

以上のデータはメイン・プログラム“GS”のすぐ後にくるようにセーブしてください。

## プログラムの起動

メイン・プログラム“GS”をロードしてRUNすると、続けてホール1のデータ“H1”を読み込み、ゲームの開始となります。

以後、各ホールごとに自動的にホール・データを読み込むのでカセット・レコーダは、PLAY状態のままにしておいてください。

## ゲーム方法

表示画面は図1のとおりです。

視野画面の緑色はフェアウェイまたはグリーン、黒がラフ、白がバンカー、木と山が紫、赤はOBです。海と池はOBと同じ処理になっています。したがって、右の全コース図では、海の部分もOBと同じ赤になっています。

OBの場合、打ち直しはしないで、最も近いフェアウェイまたはラフから2打加算して次のショットになります。

ボールの位置は、全コース図に表示されます。

ショットは次の順で行なってください。

●クラブの選択：クラブはフルセットです。

カーソル・キー $\left[ \begin{smallmatrix} \blacksquare \\ \square \end{smallmatrix} \right]$ で表示クラブが変化していくので、目的のクラブになったら $\left[ \text{RET} \right]$ でセットしてください。クラブの選択は実際のゴルフと同様に行なってください。

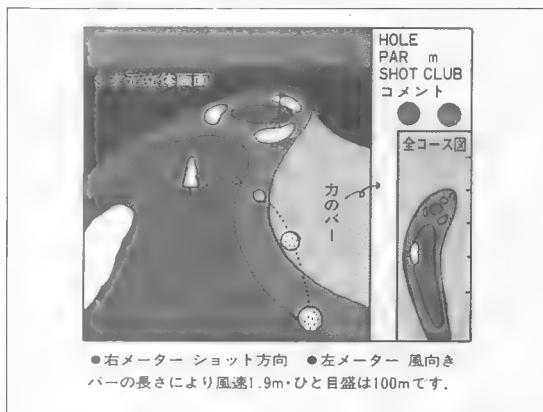
バンカーなどで、ドライバーを使ってもまともに飛ばしません。

●ショットの方向：クラブの選択を終えると、ショット方向の矢印が点滅し始めます。カーソルキー $\left[ \rightarrow \right]$ で右回りに5°づつ、 $\left[ \leftarrow \right]$ で左回りに5°づつ、 $\left[ \uparrow \right]$ で右回りに90°づつ、 $\left[ \downarrow \right]$ で左回りに90°づつ動くので目的の方向にして、 $\left[ \text{RET} \right]$ でセットしてください。

●ショット：ショット方向が決まると、力のバーが上下に動くので、うまく目的の強さで、 $\left[ \text{スペース} \right]$ を押してください。

ショットすると、視野画面の中と、全コース図の上をボールが飛んでゆきます。ボールが止まると、フェアウェイ、バンカーなどの状態がコメントとして表示されます。

図1 画面構成



視野画面が新しく描き換えられ(複雑な3次元計算をしているので、多少時間がかかります)BEEP音がすると次のショットになるので、同様の手順で行なってください。

ボールがグリーン上にあるか、ピンよりむこうにある場合は、視野画面を描き換えしないで、すぐに次のショット、(パット)になります。

なお、視野画面は、ショット位置上空40mから全コース図の上方向から40°下を見下した視野を表示しています。ボールが視野外に出た場合、または弾んで見にくい場合には、全コース図を参考にしてください。

●アウト9ホールが終了すると、アウトのスコアが表示されます。

● $\left[ \text{スペース} \right]$ を押すと、インに入ります。

●アウト、イン18ホールが終了すると、全スコアが表示されます。

● $\left[ \text{Y} \right]$ を押すと、スコア・カードがプリンタに出力されます。

●木のまわりはラフと同じ判定をします。

## 終わりに

T & E SOFTでは、FM-7用だけでなくFM-8用の“3-Dゴルフ・シミュレーション”も発売しています。また、他機種用も各機種の性能を生かしたものを発売予定しているので楽しみに。

本シミュレーションのボールの飛距離は、ごく一般のアマチュア・ゴルファーに合わせてあります。ちなみに、T & E SOFT内での最高スコアはパレーの72です。ホールインワンはまだありません。初めのうちは90以上たたくと思います。

### リスト1 “GS”プログラム・リスト

```
1 3-D GOLF SIMULATION
2 FOR FUJITSU MICRO 7
3 T&E SOFT 1983
4 By EIJI YOKODYAMA
5
6
10 H=1:PI=3.14159/180:WIDTH80,25
20 DIM DI(600),DN(600),DZ(400),X(4),Y(4),Z(4),F(
7),C$(13),C1(13),C2(13),J$(7),HO(18),C%(1):G=9.8
083:XL=540:YL=50:XE=615:YE=50:SC0=0:SC1=0
30 J$(0)="TEE SHOT":J$(1)="FAIR WAY":J$(2)="ROUG
H":J$(3)="GREEN":J$(4)="BUNKER":J$(5)="WATER HAZ
ARD":J$(6)="OB !!!":J$(7)="CUP IN !!!":LCL$=""
40 TT$=TIME$:RANDOMIZE(VAL(MID$(TT$,4,2))*60+VAL
(RIGHT$(TT$,2)))
50 SCREEN7,7:COLOR7,0:WIDTH80:CLS:C$(1)="1W":C$(
2)="3W":C$(3)="5W":C$(4)="3I":C$(5)="4I":C$(6)=""
```

```
5I":C$(7)="6I":C$(8)="7I":C$(9)="8I":C$(10)="9I"
:C$(11)="PW":C$(12)="SW":C$(13)="PT"
60 C1(1)=30:C1(2)=34:C1(3)=38:C1(4)=40:C1(5)=44:
C1(6)=46:C1(7)=48:C1(8)=52:C1(9)=56:C1(10)=60:C1
(11)=65:C1(12)=68:C1(13)=0
70 C2(1)=250:C2(2)=225:C2(3)=220:C2(4)=210:C2(5)
=205:C2(6)=200:C2(7)=195:C2(8)=185:C2(9)=180:C2
(10)=175:C2(11)=160:C2(12)=155:C2(13)=85
80 LL$=""
"ZL$="0000"
100 GOSUB2000:CLS
113 GOSUB9000:IX1=-125:IY1=600:IX2=125:IY2=0:GOS
UB7000:LINE(IX1,IY1)-(IX2,IY2),PSET,2,B
115 GOSUB4000:IX1=-125:IY1=600:IX2=125:IY2=0:GOS
UB7000
117 LINE(IX1,IY1)-(IX2,IY2),PSET,7,B:FORI=0TO600
STEP100:IX1=115:IY1=I:IX2=125:IY2=I:GOSUB7000:LI
NE(IX1,IY1)-(IX2,IY2),PSET,7:NEXTI:IX1=-10:IY1=0
```

# 3-Dゴルフ・シミュレーション

## リスト 1

```

:IX2=10: IY2=0:GOSUB7000:LINE(IX1,IY1)-(IX2,IY2),
PSET,4:READHX,HZ
120 DX=0:DY=40:DZ=1:C=-40:D=0:E=0
130 GOSUB5000
200 ***** SHOT *****
210 SC1=0:JT=0:Y0=.1:Z0=1:X0=0:PP=4:GG=0
220 GOSUB6900:SC0=SC0+1:SC1=SC1+1:COLOR7:LOCATE6
3,3:PRINTLCL$:LOCATE63,3:PRINTJ$(JT):LOCATE67,2:
PRINTSC1:GOSUB900:GOSUB1000:GOSUB6900
230 PW=PW-1:IFPW<=2THENPW=2
235 IFJT<>0ANDCL=1THENCL=2
237 IFCL=13THENV0=PW*2*C2(CL)/200:T=0:S1=C1(CL):
F0=0:S2=VV+90:GOTO250
240 GOSUB6100:V0=PW*2*C2(CL)/220+W1*COS((VV-W0)*
PI)/3:T=0:S1=C1(CL):F0=W1*SIN((VV-W0)*PI)/4:S2=V
V+90
250 IFJT=2THENV0=V0*.8:F0=F0+RND(1)-.5
260 IFJT<>4THEN300
270 IFCL<4THENF0=F0+(RND(1)-.5)*3:V0=V0*(RND(1)
*150+50)/C2(CL):GOTO300
280 IFCL=13THEN220
290 IFCL>10THENV0=V0*(RND(1)/4+.75)*C1(CL)/68:F0
=F0+(RND(1)-.5)/4:GOTO300
295 F0=F0+(RND(1)-.5)*2:V0=V0*.6
300 GOSUB6900:***** BALL *****
305 IFCL=13THENS=1:ELSES=V0/35:GOSUB6300
307 IFINT(V0)=0THENS=0
308 IFS1<0THENS1=0:T=.4:Y1=0:Z1=V0*T*COS(S2*PI)+
Z0:X1=V0*T*SIN(S2*PI)+X0:V0=V0*.85:Z0=Z1:X0=X1:G
OTO360
310 Y1=V0*SIN(S1*PI)*T-.5*G*T*T+Y0
320 Z1=V0*COS(S1*PI)*T*COS(S2*PI)+Z0
330 X1=V0*COS(S1*PI)*T*SIN(S2*PI)+X0
340 T=T+.4
350 IFY1<0THENT=0:V0=V0*.4:S1=S1*.6:Z8=Z1:X0=X1
360 XP1=INT(X1):YP1=INT(Z1)
370 XP2=Q1*(X1-DX)+Q2*(Y1-DY)+Q3*(Z1-DZ):YP2=Q4*
(X1-DX)+Q5*(Y1-DY)-Q6*(Z1-DZ):ZP2=Q7*(X1-DX)+Q8*
(Y1-DY)+Q9*(Z1-DZ)
372 XP2=INT(XP2/ZP2*240)+239:YP2=INT(YP2/ZP2*100
)+99:YP2=200-YP2
380 IX1=XP1:IY1=YP1:GOSUB7000:XXP=IX1:YYP=IY1:GO
SUB8000:PP=PCC:IFYP1>6000RYP1<0ORXP1>1250RXP1<-1
25THENBEEP:PP=2:GOTO560
383 PSET(IX1,IY1,6)
388 IFZP2<=0ORXP2>478ORXP2<10RYP2>1980RYP2<1THEN
395
387 IF Y1<0 THEN395
390 R=INT(RS/ZP2*240):IFR<=2THEN393
392 CIRCLE(XP2,YP2),R,6:PAINT(XP2,YP2),6,6:GOTO3
95
393 PSET(XP2,YP2,6)
395 IFSQR((HX-X1)^2+(HZ-Z1)^2)<.8ANDY1<.5ANDV0<3
.5THENJT=7:GOTO580
400 IFS1>=8THENS2=S2-F0
410 IFY1>5THEN307
420 IFY1<=0THEN440
430 IFPP=3THENV0=V0*.2
435 GOTO307
440 IFPP=0THENV0=V0*.6:S1=S1*.6
450 IFPP=7THEN500
455 IFPP=3THENV0=V0*.2:GOTO307
460 IFPP=10RPP=2THENV0=0:GOTO500
470 GOTO307
500 YPP=IY1:GOSUB6900
510 IFPP=6THENYPP=YPP-1:XXP=IX1:YYP=YPP:GOSUB800
0:PP=PCC:GOTO510
520 IFPP<>4THEN540ELSEJT=1
530 IFSQR((HX-X1)^2+(HZ-Z1)^2)<.50THENJT=3
540 IFPP=7THENJT=4:GOSUB6400
550 IFPP=1THENJT=5:GOTO3000
560 IFPP=2THENJT=6:GOTO3000
570 IFPP=3ORPP=0THENJT=2
580 LOCATE63,3:COLOR7:PRINTLCL$:LOCATE63,3:PRINT
J$(JT)
585 IF JT=3 OR Z1>HZ-15 THEN GG=GG+1ELSE GG=0
590 IFJT=7THENGOSUB6200:GOTO700
600 IF GG>1 THEN DX=HX:DZ=HZ-25:Y0=.1:X0=X1:Z0=Z
1:C=-46:GOTO220
610 IF JT=3 OR Z1>HZ-15 THEN DX=HX:DZ=HZ-25:Y0=.
1:X0=X1:Z0=Z1:C=-46:GOTO630
620 DX=X1:DZ=Z1:Y0=.1:X0=X1:Z0=Z1:C=-40
630 GOSUB5000:GOTO220
700 ***** SCORE CARD *****
710 HO(H)=SC1:H=H+1:IF H=10 OR H=19 THEN 730
720 GOTO100
730 LOCATE0,0:COLOR7,0:FORI=0TO23:PRINTLL$:NEXTI
:PRINTLL$:LOCATE0,1
740 OUT=0:IN=0:FORI=1TO9:OUT=OUT+HO(I):IN=IN+HO(

```

```

I+9):NEXTI
750 COLOR4,0:PRINTSPC(21)"<<< SCORE CARD >>>":PR
INT:COLOR7,0
760 PRINTSPC(5)"
"
762 PRINTSPC(5)"IHOLE 1 1 2 1 3 1 4 1 5 1 6 1 7
1 8 1 9 1 OUT 1"
764 PRINTSPC(5)"
"
766 PRINTSPC(5)"IMet. 136215081172133213581467138
21147134813076m1"
768 PRINTSPC(5)"
"
770 PRINTSPC(5)"IPAR 1 4 1 5 1 3 1 4 1 4 1 5 1 4
1 3 1 4 1 361"
772 PRINTSPC(5)"
"
774 PRINT USING " INAME### ## ## ## ## ##
## ## ## ## ## ## ## ## ## ## ## ## ## ## ##
## ## ## ## ## ## ## ## ## ## ## ## ## ## ##
HO(5);HO(6);HO(7);HO(8);HO(9);OUT
776 PRINTSPC(5)"
"
778 PRINT:IFH=19THENB00
780 COLOR5,0:PRINTSPC(5)"PUSH SPACE BAR":COLOR7,
0
790 K9$=INKEY$:IFK9$="" THEN100ELSEGOTO790
800 PRINTSPC(5)"
"
802 PRINTSPC(5)"IHOLE110 111 112 113 114 115 116
117 118 1 IN ITOTAL1"
804 PRINTSPC(5)"
"
806 PRINTSPC(5)"IMet. 137311551332144113661372132
1138148212987m16063m1"
808 PRINTSPC(5)"
"
810 PRINTSPC(5)"IPAR 1 4 1 3 1 4 1 5 1 4 1 4 1 4
1 3 1 5 1 36 1 72 1"
812 PRINTSPC(5)"
"
814 PRINT USING " INAME### ## ## ## ## ##
## ## ## ## ## ## ## ## ## ## ## ## ## ## ##
## ## ## ## ## ## ## ## ## ## ## ## ## ## ##
HO(13);HO(14);HO(15);HO(16);HO(17);HO(18);IN
SC0
816 PRINTSPC(5)"
"
820 PRINT:COLOR5,0:PRINT" PRINT OUT ? PUSH
(Y) or (N) KEY":LOCATE0,0
830 KB$=INKEY$:IFKB$="N"THENB50
840 IFKB$="Y"THEN HARDC 1:GOTO850
845 GOTO830
850 COLOR7,0:END
900 X8=X0:Y8=Y0:Z8=Z0
910 XF9=Q1*(X8-DX)+Q2*(Y8-DY)+Q3*(Z8-DZ):YP9=Q4*
(X8-DX)+Q5*(Y8-DY)-Q6*(Z8-DZ):ZP9=Q7*(X8-DX)+Q8*
(Y8-DY)+Q9*(Z8-DZ)
920 XP9=INT(XP9/ZP9*240)+239:YP9=INT(YP9/ZP9*100
)+99:YP9=200-YP9
930 IFZP9<=0ORXP9>478ORXP9<10RYP9>1980RYP9<1THEN
RETURN
940 PSET(XP9,YP9,2):RETURN
1000 CL=1:BEEP
1010 LOCATE76,2:PRINTC$(CL):K$=INKEY$:IFK$=""THE
N1010
1015 AS=ASC(K$):IF AS=&H0D THEN 1100
1020 IF AS=&H1C THEN CL=CL+1
1030 IF AS=&H1D THEN CL=CL-1
1040 IF CL=0 THEN CL=13
1050 IF CL=14 THEN CL=1
1060 GOTO1010
1100 VV=-90
1120 GOSUB6900:K$=INKEY$:IFK$=""THEN1120
1125 AS=ASC(K$):IF AS=&H0D THEN 1200
1130 IF AS=&H1C THEN VV=VV+5
1140 IF AS=&H1D THEN VV=VV-5
1142 IF AS=&H1E THEN VV=VV+90
1144 IF AS=&H1F THEN VV=VV-90
1150 IF VV=365 THEN VV=0
1160 IF VV=-5 THEN VV=355
1170 GOTO1120
1200 PW=1:UD=1
1210 LOCATE61,25-PW: IF UD=1 THEN CC=3 ELSE CC=0
1220 COLOR CC:PRINT"■":
1230 K$=INKEY$: IF K$="" THEN RETURN
1240 PW=PW+UD:IF PW=0 THEN PW=1:UD=-1
1250 IF PW=21 THEN PW=20:UD=-1
1260 GOTO1210
2000 ***READ DATA ***
2010 LOCATE0,0:FI$="H"+RIGHT$(STR$(H),LEN(STR$(H

```

# 3-Dゴルフ・シミュレーション

リスト 1

```

)) - 1:PRINTFI$
2020 OPEN "I",#1,FI$:INPUT#1,N
2030 FORI=1TON:INPUT#1,DI(I):NEXTI:CLOSE#1:RETUR
N
3000 ***** D.B *****
3010 GOSUB500:RO=10:COLOR7:LOCATE63,3:PRINTLCL$
:LOCATE63,3:PRINTJ$(JT)
3020 KO=0
3030 POZ=INT(RO*SIN(KO*PI)+Z1):POX=INT(RO*COS(KO
*PI)+X1)
3040 IX1=POX: IY1=POZ:GOSUB7000:XXP=IX1:YYP=IY1:G
OSUB8000:PP0=PC: IFPP0=0ORPP0=4THEN3045ELSEGOTO3
050
3045 IFPOZ<600ANDPOZ>0ANDPOX<125ANDPOX>-125THEN3
065
3050 KO=KO+15:IFKO<360THEN3030
3060 RO=RO+10:GOTO3020
3065 IFPP0=0THENJT=2ELSEJT=1
3070 X1=POX:Z1=POZ:SC0=SC0+2:SC1=SC1+2:PSET(IX1,
IY1,6):GOTO610
4000 ***** COURSE DISP SUB. *****
4010 I=2:CQ=2
4020 IFDI(I+3)=.5THENRETURN
4040 IFDI(I+3)>0ANDDI(I+3)<1THEN4070
4050 IX1=INT(DI(I)):IY1=INT(DI(I+2)):IX2=INT(DI(
I+3)):IY2=INT(DI(I+5)):GOSUB7000:LINE(IX1,IY1)-(
IX2,IY2),PSET,CQ
4060 I=I+3:GOTO4020
4070 ON CQ GOSUB 4100,4100,4200,4100,4100,4100,4
100,4100,4100
4080 I=I+4:GOTO4020
4100 READIX1,IY1:GOSUB7000:PAINT(IX1,IY1),CQ,CQ:
CQ=DI(I+3)*10:RETURN
4200 RETURN
5000 ***** 3-D SUB. : DX,DY,DZ : C=Xカ7,D=Yカ7,E=Z
カ7 *****
5010 I=1
5012 DN(I)=DI(I):IFDI(I)=-.97THEN5020ELSEI=I+1:GO
T05012
5020 I=1:*** DX,DY,DZ **
5025 IFDI(I)=-.97THEN5040
5026 IFDI(I)=0THEN5030
5027 IFDI(I)>-1ANDDI(I)<1THENI=I+1:GOTO5025
5030 DN(I)=DN(I)-DX:DN(I+1)=DN(I+1)-DY:DN(I+2)=D
N(I+2)-DZ
5035 I=I+3:GOTO5025
5040 *** PICTH,HEADING, BANK **
5050 Q1=COS(D*PI)*COS(E*PI)+SIN(D*PI)*SIN(E*PI)*
SIN(C*PI)
5060 Q2=SIN(D*PI)*SIN(C*PI)*COS(E*PI)-COS(D*PI)*
SIN(E*PI)
5070 Q3=SIN(D*PI)*COS(C*PI)
5080 Q4=SIN(E*PI)*COS(C*PI):Q5=COS(E*PI)*COS(C*P
I):Q6=SIN(C*PI)
5090 Q7=COS(D*PI)*SIN(C*PI)*SIN(E*PI)-SIN(D*PI)*
COS(E*PI)
5100 Q8=SIN(E*PI)*SIN(D*PI)+COS(D*PI)*SIN(C*PI):
Q9=COS(D*PI)*COS(C*PI)
5110 I=1
5115 IFDI(I)=-.97THEN5160
5116 IFDI(I)=0THEN5120
5117 IFDI(I)>-1ANDDI(I)<1THENI=I+1:GOTO5115
5120 X1=DN(I):Y1=DN(I+1):Z1=DN(I+2)
5130 DN(I)=Q1*X1+Q2*Y1+Q3*Z1
5140 DN(I+1)=Q4*X1+Q5*Y1+Q6*Z1
5145 DN(I+2)=Q7*X1+Q8*Y1+Q9*Z1
5150 I=I+3:GOTO5115
5160 LOCATE0,0:COLOR7,0:FORI=0TO23:PRINTLL$:NEXT
I:PRINTLL$:LOCATE0,0
5161 LINE(0,0)-(479,199),PSET,7,B
5164 I=2
5165 IFDI(I+3)=-.8THEN5700
5166 IFDI(I+3)=0THEN5170
5167 IFDI(I+3)>-1ANDDI(I+3)<1THENCQ=DI(I+3)*10:I
=I+4:GOTO5165
5170 X(0)=DN(I):Y(0)=DN(I+1):Z(0)=DN(I+2)
5180 X(4)=DN(I+3):Y(4)=DN(I+4):Z(4)=DN(I+5)
5190 FORA=0TO4 STEP4
5200 F(A)=0:F(A+1)=0:F(A+2)=0:F(A+3)=0
5210 IFX(A)<-Z(A) THENF(A)=1
5220 IFX(A)>Z(A) THENF(A+1)=1
5230 IFY(A)<-Z(A) THENF(A+2)=1
5240 IFY(A)>Z(A) THENF(A+3)=1
5250 NEXTA
5260 FORA=0TO3
5270 IFF(A)=0THEN5290
5280 IFF(A)=F(A+4) THEN5680
5290 NEXTA
5300 FORA=0TO3

```

```

5310 IFF(A)=1THEN5370
5320 NEXTA
5330 FORA=4TO7
5340 IFF(A)=1THEN5380
5350 NEXTA
5360 GOTO5600
5370 A=0:B=4:GOTO5390
5380 A=4:B=0
5390 IFF(A)=1THEN5480
5400 IFF(A+1)=1THEN5440
5410 IFF(A+2)=1THEN5520
5420 IFF(A+3)=1THEN5560
5430 GOTO5600
5440 K=(Z(A)-X(A))/(X(B)-X(A)-Z(B)+Z(A))
5450 X(A)=K*(Z(B)-Z(A))+Z(A)
5460 Y(A)=K*(Y(B)-Y(A))+Y(A):Z(A)=X(A)
5470 GOTO5190
5480 K=(Z(A)+X(A))/(X(A)-X(B)-Z(B)+Z(A))
5490 X(A)=K*(Z(A)-Z(B))-Z(A)
5500 Y(A)=K*(Y(B)-Y(A))+Y(A):Z(A)=-X(A)
5510 GOTO5190
5520 K=(Z(A)+Y(A))/(Y(A)-Y(B)-Z(B)+Z(A))
5530 X(A)=K*(X(B)-X(A))+X(A)
5540 Y(A)=K*(Z(A)-Z(B))-Z(A):Z(A)=-Y(A)
5550 GOTO5190
5560 K=(Z(A)-Y(A))/(Y(B)-Y(A)-Z(B)+Z(A))
5570 X(A)=K*(X(B)-X(A))+X(A)
5580 Y(A)=K*(Z(B)-Z(A))+Z(A):Z(A)=Y(A)
5590 GOTO5190
5600 IFINT(Z(0))=0THEN5680
5610 IFINT(Z(4))=0THEN5680
5620 X1=INT(X(0)/Z(0)*239)+239
5630 Y1=INT(Y(0)/Z(0)*100)+99:Y1=200-Y1
5640 X2=INT(X(4)/Z(4)*239)+239
5650 Y2=INT(Y(4)/Z(4)*100)+99:Y2=200-Y2
5670 GOSUB5690
5680 I=I+3:GOTO5165
5690 LINE(X1,Y1)-(X2,Y2),PSET,7:RETURN
5700 I=I+4
5710 IFDI(I)=-.97THENRETURN
5715 IFDI(I)=0THEN5730
5720 IFDI(I)>-1ANDDI(I)<1THENCQ=DI(I)*10:I=I+1:(
0TO5710
5725 IFDN(I+2)<0THEN5800
5730 PX=INT(DN(I)/DN(I+2)*240)+239
5740 PY=INT(DN(I+1)/DN(I+2)*100)+99:PY=200-PY
5750 IFPX>478THEN5800
5760 IFPX<1THEN5800
5770 IFPY>198THEN5800
5780 IFPY<1THEN5800
5790 PCQ=POINT(PX,PY):IF PCQ=-1 THEN 5800 ELSE P
AINT(PX,PY),CQ,7
5800 I=I+3:GOTO5710
6000 ***** SOUND SUB. *****
6100 SOUND6,15:SOUND7,7:SOUND8,16:SOUND9,16:SOU
D10,16:SOUND12,9:SOUND13,3:RETURN
6200 SOUND7,&HF8:SOUND8,10:SOUND1,3:SOUND2,20:SO
UND3,7:SOUND4,30:SOUNDS,0:SOUND6,15:SOUND8,16:SO
UND9,16:SOUND10,16:SOUND12,25:SOUND13,3:RETURN
6300 SOUND6,30:SOUND7,7:SOUND8,4:SOUND12,25:SOU
ND13,1:RETURN
6400 SOUND6,60:SOUND7,&HF7:SOUND8,16:SOUND12,3:S
OUND13,1:RETURN
6500 SOUND0,0:SOUND1,11:SOUND7,&HFE:SOUND8,16:SO
UND12,50:SOUND13,3:RETURN
6900 SOUND0,0:SOUND1,0:SOUND2,0:SOUND3,0:SOUND4,
0:SOUND5,0:SOUND6,0:SOUND7,56:SOUND8,64:SOUND9,6
4:SOUND10,64:SOUND11,255:SOUND12,255:SOUND13,0:R
ETURN
7000 IX1=INT(IX1*136/250+572):IX2=INT(IX2*136/25
0+572):IY1=INT(199-IY1*144/600):IY2=INT(199-IY2*
144/600)
7010 IFIX1>639THENIX1=639
7020 IFIX2>639THENIX2=639
7030 IFIY1>199THENIY1=199
7040 IFIY2>199THENIY2=199
7050 RETURN
8000 *****
8010 GETA(XXP,YYP)-(XXP,YYP),C%,0
8020 HE1$=HEX$(C%(0)):HE2$=HEX$(C%(1))
8030 IF LEN(HE1$)=4 THEN 8040 ELSE LA=4-LEN(HE1$
):HE1$=LEFT$(ZL$,LA)+HE1$
8040 IF LEN(HE2$)=4 THEN 8050 ELSE LB=4-LEN(HE2$
):HE2$=LEFT$(ZL$,LB)+HE2$
8050 RG1=VAL("&H"+LEFT$(HE1$,2)):RG2=VAL("&H"+RI
GHT$(HE1$,2)):RG3=VAL("&H"+LEFT$(HE2$,2))
8060 RG1=RG1 AND &H80:RG2=RG2 AND &H80:RG3=RG3 A
ND &H80
8070 IFRG1<0THENRG1=1

```

# 3-Dゴルフ・シミュレーション

## リスト1

```
8080 IFRG2<>0THENRG2=2
8090 IFRG3<>0THENRG3=4
8100 PCC=(RG1 OR RG2 OR RG3):RETURN
9000 ***** SCREEN *****
9010 LOCATE63,0:COLOR6:PRINT"HOLE"
9020 LOCATE63,1:COLOR5:PRINT"PAR"
9030 LOCATE63,2:COLOR4:PRINT"SHOT":LOCATE71,2:CO
LOR3:PRINT"CLUB"
9040 CIRCLE(532,43),24,6:CIRCLE(612,43),24,6:PAI
NT(532,43),5,6:PAINT(612,43),5,6
9050 COLOR7:LOCATE67,0:PRINTH:READP,L$:LOCATE67,
1:PRINTP "L$:W0=RD(1)*360:W1=INT(RND(1)*9+1)
9500 ***** WIND DISP SUB. *****
9510 LINE(532,43)-(XL,YL),PSET,5:XL=INT((W1*2+2)
)*COS(W0*PI)+532:YL=INT((W1*2+2)*.4495*SIN(W0*PI
)+43):LINE(532,43)-(XL,YL),PSET,2:RETURN
9600 ***** VECTOR DISP SUB. *****
9610 LINE(612,43)-(XE,YE),PSET,5:XE=20*COS(VV*PI
)+612:YE=20*SIN(VV*PI)+43:LINE(612,43)-(XE
,YE),PSET,1:RETURN
11000 DATA 4,373m,0,500,30,180,-40,350,-58,396,-
60,300,-10,180,-40,366,-40,366,-40,366
12000 DATA 5,508m,0,550,0,250,-40,400,-40,460,0,
370,-20,180,-46,410,-46,410,-46,410
13000 DATA 3,172m,0,500,-36,100,-60,160,-80,126,
-64,164,-64,164,-64,164
14000 DATA 4,333m,0,500,0,100,0,330,50,300,0,280
,-4,324,-4,324,-4,324
15000 DATA 4,358m,0,500,-30,220,0,330,-20,370,20
```

```
,270,4,328,4,328,4,328
16000 DATA 5,467m,0,500,50,200,-60,440,-80,395,-
20,215,-23,315,38,67,-64,450,-64,450
17000 DATA 4,382m,0,500,0,170,-60,380,-13,390,-7
8,326,-35,180,-60,370,-60,370,-60,370
18000 DATA 3,147m,0,500,-40,140,-25,99,-56,174,-
32,140,-32,140,-32,140
19000 DATA 4,348m,0,500,20,100,80,330,86,280,65,
155,0,180,64,326,64,326,64,326
21000 DATA 4,373m,0,500,0,60,70,388,50,340,70,33
0,70,330,60,300,12,318,-60,180,70,330
22000 DATA 3,155m,0,500,60,140,76,106,16,120,60,
140,60,140,60,140
23000 DATA 4,332m,0,500,40,140,-50,280,-15,310,3
5,250,14,190,-40,290,-40,290,-40,290
24000 DATA 5,441m,0,558,-40,300,-70,440,-26,410,
-70,340,-60,230,-60,440,-60,440,-60,440
25000 DATA 4,366m,0,500,50,200,-50,350,-50,310,9
8,190,-60,350,-60,350,-60,350
26000 DATA 4,372m,0,500,80,160,-60,250,-5,260,-5
5,294,-70,250,-70,250,-70,250
27000 DATA 4,328m,0,500,-40,170,28,310,40,348,-1
8,142,30,260,30,310,30,310,30,310
28000 DATA 3,138m,0,500,0,50,0,140,50,140,15,178
,-10,140,-10,140,-10,140
29000 DATA 5,482m,0,508,60,240,-60,430,-6,426,46
,316,-60,370,48,170,-60,430,-60,430,-60,430
```

## リスト2 \*OUT\*データ・リスト(H1~H9)

```
10 DIMDI(700)
20 I=1
30 READ DI(I):IF DI(I)=-.9 THEN 40 ELSE I=I+1:GO
TO30
40 BEEP:INPUT"FILE NAME":FI$:OPEN "O",#1,FI$:PRI
NT#1,I
50 FOR J=1 TO I:PRINT#1,DI(J):NEXTJ:CLOSE#1:GOTO
20
11000 ***** HOLE1 DATA ***
11110 DATA .2,20,0,0,74,0,140,80,0,164,78,0,190,
18,0,314,8,0,318,-10,0,320,-14,0,326,-4,0,340,0,
0,360,-14,0,400,-44,0,420,-86,0,410,-98,0,380,-9
4,0,306,-62,0,200,-44,0,140,-20,0,0,20,0,0
11120 DATA .4,4,0,10,14,0,40,60,0,160,58,0,186,2
0,0,260,-14,0,300,-28,0,298,-38,0,280,-42,0,250,
-48,0,240,-50,0,210,-38,0,160,-18,0,60,-10,0,20,
-4,0,10,4,0,10
11130 DATA .4,-36,0,330,-26,0,336,-18,0,348,-10,
0,364,-24,0,376,-40,0,382,-56,0,380,-68,0,370,-7
2,0,354,-68,0,342,-54,0,332,-36,0,330
11140 DATA .7,-30,0,394,-46,0,406,-64,0,404,-70,
0,394,-78,0,388,-84,0,380,-84,0,368,-80,0,368,-6
8,0,384,-50,0,390,-32,0,390,-30,0,394
11150 DATA .7,-60,0,274,-54,0,276,-46,0,290,-42,
0,316,-44,0,318,-56,0,310,-62,0,312,-72,0,328,-7
8,0,334,-80,0,330,-70,0,300,-68,0,284,-60,0,274
11160 DATA .7,4,0,166,6,0,186,0,0,197,-10,0,200,
-20,0,210,-28,0,208,-26,0,196,-28,0,188,-26,0,18
2,-14,0,160,-14,0,140,-12,0,138,4,0,166
11170 DATA .6,-40,0,365.5,-39.64,0,365.64,-39.5,
0,366,-39.64,0,366.36,-40,0,366.5,-40.36,0,366.3
6,-40.5,0,366,-40.36,0,365.64,-40,0,365.5
11180 DATA .6,-40,0,366,-40,5,366,-39,4,3,366,-4
0,4,366
11190 DATA .3,-30,0,210,-29,0,210,-29,3,210,-27,
2,210,-27,3,210,-29,10,210,-30,10,210,-32,3,210,
-32,2,210,-30,3,210,-30,0,210
11200 DATA .3,40,0,130,41,0,130,41,3,130,43,2,13
0,43,3,130,41,10,130,40,10,130,38,3,130,38,2,130
,40,3,130,40,0,130
11210 DATA .5,-30000,0,30000,30000,0,30000,.2,-8
6,0,418,-10000,0,-10000
11220 DATA .8,-4,20,0,180,18,0,238,-20,0,297,-3
6,0,280,-52,0,350,-28,0,350,-24,0,372,-56,0,378,
.7,-8,0,188,0,0,192,-26,0,206,-58,0,294,-46,0,31
2,-75,0,327,-34,0,393,-58,0,396,-80,0,380
11230 DATA .3,-29.5,5,210,40,5,130,.5,0,100,10
000,.1,0,0,2000,.2,-100,0,390,-110,0,370,-100,0,
260,-.9
12000 DATA .2,50,0,0,4,0,164,10,0,192,70,0,260,8
2,0,310,-10,0,460,-50,0,488,-90,0,340,-106,0,440,
-52,0,320,-44,0,292,-100,0,192,-104,0,174,-98,0,
150,-30,0,0,50,0,0
12010 DATA .4,14,0,16,26,0,40,-6,0,170,0,0,194,4
8,0,250,60,0,280,58,0,300,24,0,340,4,0,348,-4,0,
344,-6,0,300,-16,0,280,-80,0,210,-94,0,0,180,-98,0
,160,-34,0,30,-10,0,10,14,0,16
```

```
12020 DATA .4,-10,0,410,-16,0,428,-30,0,438,-48,
0,438,-60,0,430,-68,0,414,-68,0,396,-60,0,386,-4
8,0,380,-30,0,382,-14,0,394,-10,0,410
12030 DATA .7,-30,0,450,-26,0,456,-30,0,466,-40,
0,472,-64,0,460,-84,0,454,-88,0,440,-80,0,430,-7
4,0,432,-60,0,448,-50,0,452,-30,0,450
12040 DATA .7,24,0,356,30,0,364,30,0,374,20,0,39
0,16,0,406,10,0,408,4,0,404,0,0,390,-10,0,370,-4
0,0,366,-50,0,370,-60,0,372,-66,0,368,-64,0,360,
-52,0,358,-40,0,330,-32,0,328,-20,0,340,-12,0,35
4,-4,0,360,24,0,356
12050 DATA .7,-6,0,206,-2,0,214,-4,0,226,-8,0,22
6,-30,0,200,-38,0,180,-24,0,160,-22,0,130,-20,0,
140,-14,0,142,-10,0,170,-6,0,206
12060 DATA .6,-46,0,409.5,-45.64,0,409.64,-45.5,
0,410,-45.64,0,410.36,-46,0,410.5,-46.36,0,410.3
6,-46.5,0,410,-46.36,0,409.64,-46,0,409.5
12070 DATA .6,-46,0,410,-46.5,410,-47,4,3,410,-4
6,4,410
12080 DATA .3,-30,0,280,-29,0,280,-29,3,280,-27,
2,280,-27,3,280,-29,10,280,-30,10,280,-32,3,280,
-32,2,280,-30,3,280,-30,0,280
12090 DATA .3,-50,0,100,-49,0,100,-49,3,100,-47,
2,100,-47,3,100,-49,10,100,-50,10,100,-52,3,100,
-52,2,100,-50,3,100,-50,0,100
12100 DATA .5,-30000,0,30000,30000,0,30000,.2,-5
2,0,320,-10000,0,-10000
12110 DATA .8,-4,-60,0,180,20,0,320,24,0,338,4,
0,346,-30,0,410,-56,0,418,-18,0,426,-58,0,428,.7
,-22,0,180,-14,0,208,-6,0,224,-34,0,330,-62,0,36
8,14,0,378,12,0,400,-74,0,446,-40,0,460,-32,0,46
4
12120 DATA .3,-29.5,5,280,-49.5,5,100,.5,0,100,1
0000,.1,0,0,2000,.2,-100,0,130,-70,0,280,-52,0,3
16,-.9
13000 DATA .2,120,0,0,30,0,36,20,0,35,0,0,28,-8,
0,32,-30,0,90,-22,0,100,-22,0,106,-30,0,112,-30,
0,110,-42,0,120,-32,0,140,-16,0,154,-14,0,164,-3
0,0,180
13010 DATA -32,0,190,-48,0,200,-66,0,196,-84,0,1
80,-98,0,174,-104,0,156,-104,0,144,-90,0,116,-74
,0,110,-54,0,118,-46,0,116,-42,0,110,-48,0,104,-
50,0,96,-44,0,88,-34,0,86,-12,0,28,-14,0,18,-40,
0,0,120,0,0
13020 DATA .4,-32,0,94,-28,0,102,-30,0,106,-40,0
,105,-44,0,100,-44,0,96,-40,0,92,-32,0,94
13030 DATA .4,-34,0,156,-35,0,174,-44,0,184,-62,
0,183,-78,0,176,-84,0,160,-82,0,146,-72,0,136,-6
0,0,136,-44,0,120,-34,0,120
13040 DATA .7,-64,0,122,-66,0,130,-76,0,132,-86,
0,142,-90,0,162,-94,0,160,-96,0,150,-92,0,130,-8
4,0,120,-78,0,118,-64,0,122
13050 DATA .6,-64,0,163.5,-63.64,0,163.64,-63.5,
0,164,-63.64,0,164.36,-64,0,164.5,-64.36,0,164.3
6,-64.5,0,164,-64.36,0,163.64,-64,0,163.5
13060 DATA .6,-64,0,164,-64.5,164,-64,-65,4,3,164,-6
4,4,164
```



13070 DATA .3,-40,0,190,-39,0,190,-39,3,190,-37,2,190,-37,3,190,-39,10,190,-44,10,190,-42,3,190,-42,2,190,-40,3,190,-40,0,190  
 13080 DATA .5,-30000,0,30000,30000,0,30000,-3,-104,0,150,-500,0,150,-500,0,144,-104,0,144  
 13090 DATA .2,102,0,270,114,0,290,112,0,310,114,0,327,102,0,340,80,0,338,60,0,340,40,0,330,36,0,300,42,0,280,60,0,270,70,0,264,80,0,264,102,0,270  
 13100 DATA .4,5,11,300,60,0,285,84,0,282,100,0,294,102,0,314,94,0,326,70,0,330,52,0,316,50,0,300,13110 DATA -.8,-4,-36,0,100,-70,0,150,-50,0,170,-40,0,178,-76,0,174,-7,-80,0,128,-67,0,128,-92,0,158,-3,-39,5,190,-5,0,100,10000,1,0,0,2000,-70,0,90,-50,0,114,-106,0,140,-4,76,0,306,-9  
 14000 DATA .2,30,0,0,60,0,114,84,0,140,100,0,170,100,0,186,86,0,208,86,0,220,90,0,236,80,0,300,4,0,330,20,0,360,-10,0,364,-28,0,352,-36,0,330,-10,0,220,0,0,190,10,0,186,10,0,180,-2,0,160,-30,0,150,-40,0,136,-30,0,0,30,0,0  
 14010 DATA .4,0,0,10,10,0,10,28,0,20,40,0,60,50,0,120,74,0,140,90,0,170,90,0,180,78,0,200,74,0,210,78,0,240,66,0,266,34,0,278,20,0,270,0,0,240,4,0,202,18,0,196,20,0,184,8,0,156,-14,0,15,30,0,130,-28,0,40,-20,0,20,0,0,10  
 14020 DATA .4,0,0,300,16,0,304,28,0,318,28,0,334,18,0,350,6,0,356,-12,0,352,-24,0,338,-26,0,320,-18,0,306,0,0,300  
 14030 DATA .7,64,0,0,280,68,0,284,62,0,300,52,0,312,42,0,314,36,0,310,36,0,296,46,0,284,64,0,280  
 14040 DATA .7,0,0,260,10,0,278,10,0,284,4,0,290,-2,0,290,-12,0,294,-16,0,290,-16,0,278,-10,0,264,-4,0,260,0,0,260  
 14050 DATA .6,-4,0,323.5,-3.64,0,323.64,-3.5,0,324,-3.64,0,324.36,-4,0,324.5,-4.36,0,324.36,-4.5,0,324,-4.36,0,323.64,-4,0,323.5,-6,-4,0,324,-4,5,324,-3,4,3,324,-4,4,324  
 14060 DATA .3,20,10,200,21,0,200,21,3,200,23,2,20,0,23,3,200,21,10,200,20,10,200,18,3,200,18,2,200,20,3,200,20,0,200  
 14070 DATA .3,50,11,180,51,0,180,51,3,180,53,2,180,53,3,180,51,10,180,58,10,180,48,3,180,48,2,180,50,3,180,50,0,180  
 14080 DATA .5,-30000,0,30000,30000,0,30000,-2,20,0,360,-30000,0,30000,-2,0,0,190,-30000,11,5000  
 14090 DATA -.8,-4,50,0,170,10,0,230,40,0,250,64,0,264,22,0,270,12,0,326,-16,0,326,-20,0,336,26,0,334  
 14100 DATA .7,52,0,0,296,38,0,308,52,0,310,-4,0,278,-12,0,292,8,0,284,-3,20,5,200,50,5,180,-5,0,100,10000,-1,0,0,2000,-7,0,0,190,8,0,286,0,0,180,-2,-60,0,360,8,0,365,-20,0,370,-9  
 15000 DATA .2,30,0,0,26,10,130,44,0,160,5,0,200,44,0,220,42,0,270,62,0,318,60,0,340,20,0,386,-20,0,396,-44,0,378,-48,0,344,-30,0,311,-28,0,330,-50,0,280,-70,0,240  
 15010 DATA .90,0,140,-72,0,144,-48,0,164,-16,0,194,0,0,198,2,0,196,0,0,188,-30,0,154,-69,0,104,-86,0,90,-30,0,0,30,0,0  
 15020 DATA .4,0,0,10,14,0,20,20,0,40,6,0,100,11,0,140,32,0,180,30,0,210,0,0,250,-8,0,280,-22,0,280,-46,0,264,-62,0,230,-72,0,144,-48,0,164,-16,0,194,0,0,198,2,8,196,0,0,188,-30,0,154,-68,0,104,-42,0,36,-20,0,14,0,0,10  
 15030 DATA .4,40,0,320,38,0,332,28,0,346,12,0,354,0,0,354,-16,0,346,-20,0,330,-14,0,316,-4,0,308,10,0,304,18,0,304,30,0,306,36,0,312,40,0,320  
 15040 DATA .7,20,0,368,22,0,372,2,0,384,-14,0,386,-34,0,370,-38,0,350,-34,0,332,-30,0,330,-24,0,350,-12,0,364,20,0,368  
 15050 DATA .7,30,0,256,36,0,280,34,0,290,26,0,292,12,0,290,6,0,286,10,0,264,14,0,250,24,0,238,28,0,230,32,0,232,30,0,256  
 15060 DATA .6,4,0,327.5,4,36,0,327.64,4,5,0,328,4,36,0,328.36,4,3,0,328,3,64,0,327.64,4,0,327.5,6,4,0,328,4,5,328,5,4,3,328,4,4,328  
 15070 DATA .3,-50,0,170,-49,0,170,-49,3,170,-47,2,170,-47,3,170,-49,10,170,-50,10,170,-52,3,170,-52,2,170,-50,3,170,-50,0,170  
 15080 DATA .5,-30000,0,30000,30000,0,30000,-2,2,0,386,-30000,0,30000,-2,-48,0,344,-30000,0,1500,0,-2,-86,0,90,-30000,0,-30000  
 15090 DATA -.8,-4,-10,0,140,28,0,208,-20,0,250,-44,0,262,-9,0,277,22,0,324,-6,0,322,36,0,330,-14,0,342,-60,0,166,-7,-20,0,370,12,0,372,20,0,371,22,0,270,14,0,288,32,0,288  
 15100 DATA .3,-49,5,170,-5,0,100,10000,-1,0,0,2000,-50,0,340,-70,0,300,-3,0,144,-80,0,120,-22,

0,154,-2,0,0,398,-80,0,420,-86,0,84,-9  
 16000 DATA .2,40,0,0,100,0,180,114,0,250,102,0,276,84,0,290,82,0,290,80,0,276,84,0,266,92,0,260,94,0,250,90,0,234,80,0,230,68,0,228,62,0,232,58,0,244,60,0,260,66,0,270,74,0,280,73,0,300,0,0,370  
 16010 DATA .20,0,420,-34,0,470,-60,0,486,-80,0,484,-100,0,460,-110,0,420,-106,0,386,-80,0,340,-50,0,260,-30,0,180,-20,0,100,-20,0,0,40,0,0  
 16020 DATA .4,0,0,10,10,0,10,40,0,40,84,0,160,94,0,214,88,0,220,60,0,222,52,0,232,50,0,256,62,0,280,64,0,290,20,0,320,-20,0,380,-40,0,392,-52,0,392,-62,0,382,-60,0,350,-36,0,280,-6,0,240,8,0,190,4,0,150,-14,0,80,-14,0,36,0,0,10  
 16030 DATA .4,-44,0,420,-34,0,430,-34,0,442,-44,0,460,-58,0,470,-70,0,470,-84,0,458,-90,0,440,-90,0,426,-80,0,420,-62,0,416,-84,0,420  
 16040 DATA .7,-80,0,378,-62,0,390,-52,0,406,-54,0,408,-64,0,404,-70,0,410,-80,0,410,-90,0,408,-96,0,420,-98,0,420,-102,0,404,-94,0,386,-80,0,378  
 16045 DATA .7,0,0,180,-8,0,220,-22,0,248,-42,0,274,-44,0,270,-36,0,240,-30,0,204,-18,0,184,-14,0,156,-10,0,154,0,0,180  
 16050 DATA .1,-10,0,318,-12,0,332,-22,0,338,-34,11,334,-38,0,320,-30,0,292,-26,0,288,-22,0,288,-20,0,296,-14,0,306,-10,0,318  
 16060 DATA .7,38,0,60,44,0,70,48,0,84,46,0,86,40,0,84,32,0,76,28,0,64,28,0,56,30,0,54,38,0,60  
 16065 DATA .6,-64,0,449.5,-63.64,0,449.64,-63.5,0,450,-63.64,0,450.36,-64,0,450.5,-64.36,0,450.36,-64.5,0,450,-64.36,0,449.64,-64,0,449.5,-6,-64,0,450,-64.5,450,-65,4,3,450,-64,4,450  
 16070 DATA .5,-10000,0,10000,10000,0,10000,-3,-4000,0,10000,0,900,10000,9000,0,10000,-2,0,0,370,20,0,420,10000,0,10000  
 16080 DATA -.8,-4,40,0,200,90,0,214,60,0,288,6,0,320,-20,0,374,-52,0,390,-50,0,440,-74,0,434,-44,0,457,-11,0,458  
 16090 DATA .7,-78,0,394,-57,0,403,-97,0,416,-20,0,214,-42,0,270,-26,0,244,36,0,68,34,0,76,46,0,114,5,-4000,200,10000,6000,500,10000,-3,0,200,10000,-2,0,0,1000,-1,100,0,400,24,0,420,72,0,245,-24,0,314,-32,0,332,-14,0,330,-9  
 17000 DATA .2,40,0,0,32,0,120,50,0,192,50,0,212,-6,0,280,-16,0,310,2,0,358,2,0,398,-10,0,426,-24,0,434,-86,0,420,-110,0,390,-114,0,360,-86,0,300,-70,0,250,-80,0,200,-88,0,160,-60,0,100,-30,0,0,40,0,0  
 17010 DATA .4,6,0,10,26,0,36,18,0,120,22,0,140,36,0,194,32,0,210,4,0,240,-24,0,280,-46,0,300,-54,0,298,-60,0,280,-60,0,212,-74,0,180,-74,0,160,-50,0,100,-28,0,32,-10,0,10,6,0,10  
 17020 DATA .4,-60,0,350,-44,0,354,-30,0,366,-28,0,384,-34,0,402,-50,0,412,-70,0,410,-86,0,398,-90,0,376,-86,0,360,-76,0,352,-60,0,350  
 17030 DATA .7,-20,0,362,-16,0,360,-6,0,378,-8,0,398,-14,0,414,-30,0,420,-32,0,414,-22,0,400,-11,0,380,-20,0,362  
 17040 DATA .7,-70,0,310,-64,0,318,-64,0,328,-68,0,334,-80,0,340,-96,0,362,-100,0,380,-104,0,366,-100,0,340,-80,0,312,-70,0,310  
 17050 DATA .7,-40,0,120,-30,0,114,-20,0,116,-14,0,128,-12,0,142,-20,0,150,-26,0,154,-26,0,160,-20,0,170,-18,0,186,-28,0,198,-38,0,200,-50,0,194,-54,0,180,-50,0,166,-40,0,156,-40,0,148,-44,0,134,-40,0,120  
 17060 DATA .6,-60,0,369.5,-59.64,0,369.64,-59.5,0,370,-59.64,0,370.36,-60,0,370.5,-60.36,0,370.36,-60.5,0,370,-60.36,0,369.64,-60,0,369.5,-6,-60,0,370,-60,5,370,-59,4,3,370,-60,4,370  
 17070 DATA .3,-30,0,300,-29,0,300,-29,3,300,-27,2,300,-27,3,300,-29,10,300,-30,10,300,-32,3,300,-32,2,300,-30,3,300,-30,0,300  
 17080 DATA .3,-50,0,200,-49,0,200,-49,3,200,-47,2,200,-47,3,200,-49,10,200,-50,10,200,-52,3,200,-52,2,200,-50,3,200,-50,0,200  
 17090 DATA .5,-30000,0,30000,30000,0,30000,-2,-2,4,0,434,60,0,470,5000,0,1000,-2,-86,0,420,-100,0,0,2,-16,0,310,40,0,360,60,0,360,1000,0,0,2,50,0,212,90,0,180,1000,0,180  
 17100 DATA -.8,-4,0,0,170,30,0,208,-58,0,200,-50,0,296,-40,0,380,-72,0,366,-40,0,400,-82,0,394,-7,-12,0,384,-28,0,414,-78,0,326,-68,0,330,-100,0,368,-28,0,132,-14,0,140,-36,0,180,-20,0,184,-48,0,192  
 17110 DATA .3,-29,5,5,300,-49,5,5,200,-5,0,100,10000,-1,0,0,2000,60,0,288,-13,0,310,-2,60,0,180,54,0,204,-100,0,140,-100,0,310,-103,0,406,60,0,420,-10,0,432,-9

# 3-Dゴルフ・シミュレーション

リスト 2

```
18000 DATA .2,10,8,108,12,0,110,10,0,144,0,0,156
,-20,0,190,-48,0,190,-74,0,178,-86,0,150,-64,0,1
00,-42,0,84,-20,0,82,10,0,100
18010 DATA .4,-24,0,120,-14,0,132,-14,0,148,-20,
0,160,-36,0,166,-52,0,162,-64,0,148,-66,0,134,-5
6,0,120,-40,0,114,-24,0,120
18020 DATA .7,-20,0,92,-8,0,104,-4,0,118,-6,0,13
4,-8,0,134,-14,0,116,-28,0,106,-40,0,116,-52,0,1
12,-54,0,110,-50,0,98,-36,0,90,-20,0,92
18030 DATA .7,-28,0,178,-36,0,182,-50,0,182,-64,
0,176,-70,0,162,-70,0,156,-62,0,160,-52,0,170,-3
8,0,172,-28,0,178
18040 DATA .6,-32,0,139.5,-31.64,0,139.64,-31.5,
0,140,-31.64,0,140.36,-32,0,140.5,-32.36,0,140.3
6,-32.5,0,140,-32.36,0,139.64,-32,0,139.5,-6,-32
0,140,-32.5,140,-33,4,3,140,-32,4,140
18050 DATA .3,-30,0,190,-29,0,190,-29,3,190,-27,
2,190,-27,3,190,-29,10,190,-30,10,190,-32,3,190,
-32,2,190,-30,3,190,-30,0,190
18060 DATA .3,0,0,100,1,0,100,1,3,100,3,2,100,3,
3,100,1,10,100,0,10,100,-2,3,100,-2,2,100,0,3,10
0,0,0,100
18070 DATA .5,-30000,0,30000,30000,0,30000,-2,-8
6,0,150,-92,0,174,-92,0,190,-56,0,250,-54,0,276,
-42,0,304,-46,0,314,-56,0,312,-300,0,0,2,10,0,1
00,40,0,124,50,0,126,56,0,116,54,0,116
18080 DATA .6,0,80,74,0,74,78,0,60,70,0,50,20,0,
40,-26,0,20,-40,0,0,2,-92,0,180,-200,0,180,-2,
92,0,174,-200,0,174,3,-5000,0,8000,0,1000,8000,
5000,0,8000
18090 DATA -.8,.4,-40,0,140,-22,0,156,-60,0,148,
.7,-24,0,100,-7,0,128,-51,0,109,-56,0,134,-32,0,
178,-66,0,162,3,-29.5,5,190,1.5,100,0,500,8000,
.5,-3500,550,8000,3500,550,8000,-2,0,0,2000,0,0,
20,-1,-20,0,60,-96,0,170,-90,0,240,-9
19000 DATA .2,40,0,0,50,0,60,50,0,70,80,0,138,90
0,160,90,0,240,100,0,266,116,0,294,112,0,340,10
0,0,360,76,0,366,50,0,350,40,0,316,52,0,290,50,1

```

```
,282,-20,0,220,-36,0,178,-30,0,100,-20,0,0,40,0,
0
19010 DATA .4,0,0,12,10,0,10,30,0,20,40,0,40,40,
0,140,44,0,170,50,0,180,70,0,176,78,0,220,78,0,2
40,64,0,272,56,0,274,40,0,264,-14,0,210,-26,0,17
0,-20,0,130,-14,0,30,0,0,12
19020 DATA .4,60,0,310,70,0,304,84,0,304,98,0,31
2,102,0,328,98,0,340,90,0,349,74,0,352,60,0,344,
54,0,326,60,0,310
19030 DATA .7,80,0,258,84,0,258,100,0,280,104,0,
304,102,0,306,88,0,294,74,0,290,72,0,280,78,0,26
8,80,0,258
19040 DATA .7,50,0,160,48,0,120,52,0,110,60,0,13
0,70,0,140,82,0,160,82,0,188,76,0,190,56,0,174,5
0,0,160
19050 DATA .1,0,0,146,10,0,180,14,0,210,18,0,224
16,0,228,10,0,224,-10,0,200,-14,0,180,-14,0,168
-6,0,144,0,0,146
19060 DATA .6,64,0,325.5,64.36,0,325.64,64.5,0,3
26,64.36,0,326.36,64,0,326.5,63.64,0,326.36,63.5
0,326,63.64,0,325.64,64,0,325.5,-6,64,0,326,64,
5,326,65,4,3,326,64,4,326
19070 DATA .3,40,0,180,41,0,180,41,3,180,43,2,18
0,43,3,180,41,10,180,40,10,180,38,3,180,38,2,180
,40,3,180,40,11,180
19080 DATA .3,-10,0,70,-9,0,70,-9,3,70,-7,2,70,-
7,3,70,-9,10,70,-10,10,70,-12,3,70,-12,2,70,-10,
3,70,-10,8,70
19090 DATA .5,-30000,0,30000,30000,0,30000,-2,10
0,0,360,10000,0,10000
19100 DATA -.8,.4,30,0,180,60,0,270,80,0,320,60,
0,340,96,0,338,-7,86,0,280,101,0,300,77,0,287,60
0,150,59,0,171,79,0,185,-1,0,180,15,0,224
19110 DATA .3,40,5,5,180,-9,5,5,70,-5,0,100,1000
0,-1,0,0,2000,-2,80,0,110,110,0,170,106,0,359,11
4,0,360,-9

```

リスト 3 INデータ・リスト (H10~H18)

```
10 DIMDI(700)
20 I=1
30 READ DI(I):IF DI(I)=-9 THEN 40 ELSE I=I+1:GOTO
T030
40 BEEP:INPUT"FILE NAME":FI$:OPEN "O",#1,FI$:PRI
NT#1,I
50 FOR J=1 TO I:PRINT#1,DI(J):NEXTJ:CLOSE#1:GOTO
20
21000 ***** HOLE10 DATA *****
21010 DATA .2,28,0,0,36,0,40,8,0,180,8,0,192,70,
0,280,100,0,340,104,0,384,70,0,410,22,0,386,-60,
0,280,-96,0,216,-100,0,170,-70,0,110,-30,0,0,28,
0,0
21020 DATA .4,8,0,10,16,0,36,20,0,60,-8,0,170,-1
0,0,190,-4,0,210,32,0,260,34,0,274,24,0,294,0,0,
296,-20,0,290,-52,0,260,-76,0,222,-80,0,200,-84,
0,180,-80,0,160,-52,0,100,-26,0,20,-10,0,10,8,0,
10
21030 DATA .7,84,0,358,88,0,366,82,0,380,72,0,39
0,60,0,388,54,0,386,50,0,380,52,0,372,62,0,372,7
0,0,368,78,0,358,84,0,358
21040 DATA .4,72,0,320,82,0,332,80,0,346,68,0,36
0,52,0,366,40,0,366,30,0,362,24,0,352,28,0,336,4
0,0,322,50,0,318,72,0,320
21050 DATA .6,70,5,0,330,70,36,0,330,36,70,0,330
.5,69.64,0,330.64,69.5,0,330,69.64,0,329.64,70,0
,329.5,70.36,0,329.64,70.5,0,330
21060 DATA .6,70,0,0,330,70,5,330,71,4,3,330,70,4,
330
21070 DATA .7,60,0,290,70,0,294,76,0,300,76,0,31
0,72,0,314,68,0,314,54,0,310,48,0,310,42,0,312,3
8,0,310,38,0,300,48,0,290,60,0,290
21080 DATA .7,14,0,300,22,0,310,20,0,320,24,0,33
0,24,0,340,20,0,342,12,0,338,4,0,324,2,0,310,6,0
,300,14,0,300
21090 DATA .7,-56,0,156,-50,0,158,-46,0,166,-50,
0,180,-50,0,178,-58,0,208,-68,0,207,-76,0,200,-7
8,0,182,-72,0,166,-62,0,156,-56,0,156
21100 DATA .3,-7,0,75,-7,3,75,-5,2,75,-5,3,75,-7
,10,75,-8,10,75,-10,3,75,-10,2,75,-8,3,75,-8,0,7
5,-7,0,75
21110 DATA .3,-17,0,225,-17,3,225,-15,2,225,-15,
3,225,-17,10,225,-18,10,225,-20,3,225,-20,2,225,
-18,3,225,-18,0,225,-17,0,225
21120 DATA .5,-30000,0,30000,30000,0,30000,-2,70
0,0,280,10000,0,-10000,-2,-60,0,280,-10000,0,-100
00
21130 DATA -.8,.4,20,0,290,-30,0,190,20,0,280,52

```

```
,0,342,-7,70,0,380,60,0,302,12,0,318,-62,0,182,-
5,0,100,10000,-1,0,0,2000,-2,70,0,120,-120,0,180
,70,0,260,-100,0,230,-3,7,5,5,75
21140 DATA -17.5,5,225,4,76,0,337,-7,-64,0,206,
20,0,340,52,0,380,40,0,308,4,30,0,356,-9
22000 ***** HOLE11 DATA *****
22010 DATA .2,90,0,90,118,0,120,120,0,140,114,0,
168,92,0,186,72,0,188,50,0,178,34,0,180,16,0,180
,4,0,163,-2,0,140,0,0,120,10,0,96,30,0,98,60,0,8
8,90,0,90
22020 DATA .4,94,0,136,96,0,146,88,0,162,70,0,17
0,50,0,168,32,0,162,26,0,150,26,0,140,34,0,128,4
6,0,120,60,0,120,80,0,124,94,0,136
22030 DATA .7,90,0,100,92,0,114,92,0,120,86,0,12
2,78,0,116,70,0,114,60,0,114,54,0,108,54,0,98,62
0,92,70,0,94,82,0,92,90,0,100
22040 DATA .7,20,0,104,32,0,110,34,0,116,24,0,12
6,18,0,142,16,0,142,10,0,136,6,0,120,10,0,110,20
,0,104
22050 DATA .6,60,0,139.5,60.36,0,139.64,60.5,1
40,60.36,0,140.36,60,0,140.5,59.64,0,140.36,59.5
,0,140,59.64,0,139.64,60,0,139.5
22060 DATA .6,60,0,140,60,5,140,61,4,3,140,60,4,
140
22070 DATA .3,22,0,180,23,0,180,23,3,180,25,2,18
0,25,3,180,23,10,180,22,10,180,20,3,180,20,2,180
,22,3,180,22,0,180
22080 DATA .5,-30000,0,30000,30000,0,30000,-2,90
0,90,10000,0,-10000,-2,114,0,168,10000,0,10000
22090 DATA -.8,.4,60,0,124,90,0,140,30,0,146,-7,
76,0,106,88,0,118,60,0,112,16,0,120,16,0,138,30,
0,114,-2,120,0,166,-3,22,5,180,-5,0,100,10000,
-1,0,0,2000,-2,116,0,167,-1,30,0,96,-9
23000 ***** HOLE12 DATA *****
23010 DATA .2,30,0,0,80,0,100,94,0,180,94,0,206,
60,0,260,-4,0,338,-30,0,336,-74,0,320,-90,0,280,
-70,0,240,-34,0,220,-16,0,140,-20,0,0,30,0,0
23020 DATA .4,8,0,20,10,0,20,40,8,40,74,130,86
0,190,84,0,210,52,0,260,30,0,268,10,0,260,-8,0,
246,-14,0,220,-16,0,176,-8,0,160,-4,0,134,-10,0,
40,0,0,20
23030 DATA .4,-60,0,300,-72,0,290,-76,0,274,-76,
0,266,-70,0,254,-58,0,250,-44,0,250,-30,0,262,-2
0,0,280,-22,0,292,-30,0,304,-44,0,316,-60,0,300
23040 DATA .7,-2,0,300,-4,0,314,-14,0,324,-30,0,
322,-46,0,312,-44,0,310,-32,0,316,-20,0,304,-16,
0,290,-16,0,280,-10,0,282,-2,0,300
23050 DATA .7,20,0,250,20,0,244,24,0,240,30,0,23

```



8,40,0,234,46,0,234,50,0,242,46,0,256,30,0,262,2  
2,0,258,20,0,250  
23060 DATA .1,0,0,222,-8,0,204,-8,0,196,-4,0,186  
-2,0,170,6,0,160,20,0,158,30,0,160,36,0,170,34,  
0,176,38,0,184,38,0,194,30,0,202,20,0,208,12,0,2  
20,6,0,220,0,0,222  
23070 DATA .6,-40,0,289.5,-39.64,0,289.64,-39.5,  
0,290,-39.64,0,290.36,-40,0,290.5,-40.36,0,290.3  
6,-40.5,0,290,-40.36,0,289.64,-40,0,289.5  
23080 DATA .6,-40,0,290,-40.5,290,-39.4,3,290,-4  
0,4,290  
23090 DATA .3,42,0,120,43,0,120,43,3,120,45,2,12  
0,45,3,120,43,10,120,42,10,120,40,3,120,40,2,120  
42,3,120,42,0,120  
23100 DATA .5,-30000,0,30000,30000,0,30000,.2,-1  
0000,0,30000,0,1300,30000,35000,0,30000  
23110 DATA -.8,-4,60,0,240,-30,0,290,-50,0,275,-  
70,0,280,20,0,120,30,0,265,0,0,250,-7,34,0,248,4  
4,0,254,24,0,256,-12,0,310,-44,0,312,.1,14,0,190  
0,0,219,30,0,200,-3,42.5,5,120,-5,-10000,300,30  
000,.3,0,800,30000,.2,0,0,1000,-.9  
24000 '\*\*\* HOLE13 DATA \*\*\*  
24010 DATA .2,30,0,0,40,0,70,26,0,220,12,0,280,0  
0,450,-26,0,482,-68,0,500,-90,0,500,-114,0,478,  
-116,0,400,-110,0,260,-70,0,100,-20,0,0,30,0,0  
24020 DATA .4,10,0,10,24,0,60,14,0,200,0,8,260,-  
6,0,330,-18,0,362,-36,0,380,-60,0,380,-86,0,372,  
-100,0,328,-90,0,230,-70,0,140,-10,0,10,10,0,10  
24030 DATA .4,-40,0,450,-50,0,466,-62,0,472,-80,  
0,470,-76,0,460,-100,0,450,-100,0,440,-94,0,420,  
-82,0,410,-66,0,408,-52,0,416,-42,0,430,-40,0,45  
0  
24040 DATA .7,-8,0,394,-10,0,410,-16,0,420,-14,0,  
436,-20,0,448,-30,0,450,-36,0,442,-36,0,430,-46  
0,410,-46,0,400,-40,0,390,-30,0,382,-18,0,382,-  
12,0,384,-8,0,394  
24050 DATA .7,-50,0,342,-52,0,356,-64,0,362,-70,  
0,360,-80,0,354,-88,0,340,-90,0,330,-88,0,314,-8  
0,0,316,-70,0,322,-58,0,330,-50,0,342  
24060 DATA .1,-20,0,220,-20,0,230,-30,0,246,-46,  
0,254,-60,0,266,-72,0,260,-80,0,258,-86,0,248,-8  
4,0,230,-76,0,214,-68,0,198,-54,0,194,-44,0,200,  
-26,0,204,-20,0,220  
24070 DATA .6,-60,0,439.5,-59.64,0,439.64,-59.5,  
0,440,-59.64,0,440.36,-60,0,440.5,-60.36,0,440.3  
6,-60.5,0,440,-60.36,0,439.64,-60,0,439.5  
24080 DATA .6,-60,0,440,-60.5,440,-61,4.3,440,-6  
0,4,440  
24090 DATA .3,-38,0,140,-37,0,140,-37,3,140,-35,  
2,140,-35,3,140,-37,10,140,-38,10,140,-40,3,140,  
-40,2,140,-38,3,140,-38,0,140  
24100 DATA .3,-8,0,240,-7,0,240,-7,3,240,-5,2,24  
0,-5,3,240,-7,10,240,-8,10,240,-10,3,240,-10,2,2  
40,-8,3,240,-8,0,240  
24110 DATA .5,-30000,0,30000,30000,0,30000,.2,-1  
0000,0,-10000,-800,0,1000,400,0,1000,10000,0,-10  
000  
24120 DATA -.8,-4,-40,0,300,-70,0,440,-48,0,457,  
-94,0,458,-60,0,376,-20,0,360,-92,0,340,-20,0,16  
0,.1,-50,0,230,-30,0,242,-82,0,248  
24130 DATA .7,-26,0,410,-20,0,444,-32,0,444,-70,  
0,340,-54,0,354,-80,0,350,-6,-60,0,440,-60,0,440  
,.3,-37.5,5,140,-7.5,5,240,.5,0,100,10000,.1,0,0  
25000,.2,0,0,750,-.9  
25000 '\*\*\* HOLE14 DATA \*\*\*  
25010 DATA .2,30,0,0,100,0,140,110,0,180,108,0,2  
10,10,0,360,-16,0,390,-50,0,400,-74,0,390,-90,0,  
370,-90,0,340,-60,0,280,-20,0,230,6,0,210,16,0,1  
92,16,0,170,0,0,144,-20,0,120,-30,0,80,-30,0,40,  
-20,0,0,30,0,0  
25020 DATA .4,20,0,10,60,0,100,80,0,180,80,0,200  
72,0,220,30,0,290,-4,0,310,-20,0,306,-28,0,292,  
-26,0,270,6,0,220,22,0,196,22,0,170,-4,0,120,-14  
0,60,0,0,10,20,0,10  
25030 DATA .4,-14,0,354,-20,0,374,-34,0,384,-56,  
0,384,-72,0,376,-80,0,360,-80,0,346,-70,0,330,-5  
6,0,324,-34,0,326,-20,0,338,-14,0,354  
25040 DATA .7,-50,0,300,-38,0,306,-34,0,318,-36,  
0,320,-58,0,318,-66,0,320,-76,3,328,-76,0,320,-7  
0,0,310,-58,0,300,-50,0,300  
25050 DATA .7,100,0,170,104,0,180,104,0,190,102,  
0,200,88,0,220,84,0,218,84,0,212,86,0,206,84,0,1  
92,84,0,182,90,0,170,94,0,168,100,0,170  
25060 DATA .6,-60,0,349.5,-59.64,0,349.64,-59.5,  
0,350,-59.64,0,350.36,-60,0,350.5,-60.36,0,350.3  
6,-60.5,0,350,-60.36,0,349.64,-60,0,349.5  
25070 DATA .6,-60,0,350,-60.5,350,-61,4.3,350,-6  
0,4,350  
25080 DATA .3,32,0,200,33,0,200,33,3,200,33,2,20

0,35,3,200,33,10,200,32,10,200,30,3,200,30,2,200  
32,3,200,32,0,200  
25090 DATA .3,82,0,230,83,0,230,83,3,230,85,2,23  
0,85,3,230,83,10,230,82,10,230,80,3,230,80,2,230  
82,3,230,82,0,230  
25100 DATA .5,-30000,0,30000,30000,0,30000,.2,-5  
0,0,400,30000,0,30000  
25110 DATA -.8,-4,-36,0,354,-70,0,355,-22,0,372,  
40,0,140,30,0,280,-4,0,307,-18,0,304,-7,-56,0,31  
0,-38,0,316,-74,0,324,92,0,188,88,0,218,.3,32.5,  
5,200,82.5,5,230,-5,0,100,10000,.1,0,0,2000,10,0  
198,-2,-10,0,430,40,0,350,114,0,130,-.9  
26000 '\*\*\* HOLE15 DATA \*\*\*  
26010 DATA .2,40,0,0,100,0,80,120,0,140,120,0,20  
0,90,0,240,-40,0,310,-80,0,300,-96,0,260,-80,0,2  
20,-40,0,170,20,0,176,34,0,160,-20,0,0,40,0,0  
26020 DATA .4,30,0,10,60,0,50,100,0,120,114,0,160  
114,0,190,80,0,236,30,0,250,-10,0,230,-40,0,190  
-36,0,180,0,0,180,30,0,186,48,0,170,46,0,150,0,  
0,40,6,0,16,20,0,10,30,0,10  
26030 DATA .4,-60,0,224,-46,0,230,-32,0,242,-28,  
0,260,-36,0,278,-52,0,284,-74,0,276,-84,0,258,-8  
2,0,240,-72,0,228,-60,0,224  
26040 DATA .7,10,0,260,8,0,266,-4,0,274,-22,0,27  
6,-26,0,270,-22,0,260,-20,0,248,-14,0,246,4,0,25  
2,10,0,260  
26050 DATA .7,-34,0,286,-40,0,298,-56,0,302,-70,  
0,294,-76,0,282,-76,0,278,-64,0,286,-46,0,288,-3  
8,0,284,-34,0,286  
26060 DATA .6,-70,0,249.5,-69.64,0,249.64,-69.5,  
0,250,-69.64,0,250.36,-70,0,250.5,-70.36,0,250.3  
6,-70.5,0,250,-70.36,0,249.64,-70,0,249.5  
26070 DATA .6,-70,0,250,-70.5,250,-69,4.3,250,-7  
0,4,250  
26080 DATA .3,32,0,170,33,0,170,33,3,170,35,2,17  
0,35,3,170,33,10,170,32,10,170,30,3,170,30,2,170  
32,3,170,32,0,170  
26090 DATA .3,42,0,160,43,0,160,43,3,160,45,2,16  
0,45,3,160,43,10,160,42,10,160,40,3,160,40,2,160  
42,3,160,42,0,160  
26100 DATA .3,82,0,180,83,0,180,83,3,180,85,2,18  
0,85,3,180,83,10,180,82,10,180,80,3,180,80,2,180  
82,3,180,82,0,180  
26110 DATA .5,-30000,0,30000,30000,0,30000,.2,-4  
0,0,310,2000,0,500  
26120 DATA -.8,-4,80,0,160,80,0,230,30,0,246,-14  
0,220,-38,0,274,-56,0,254,-76,0,244,-7,-10,0,26  
0,6,0,264,-22,0,272,-56,0,294,-36,0,287,.3,32.5,  
5,170,42.5,5,160,82.5,5,180  
26130 DATA .5,0,100,10000,.1,0,0,2000,24,0,164,.  
2,120,0,120,50,0,300,-10,0,310,-.9  
27000 '\*\*\* HOLE16 DATA \*\*\*  
27010 DATA .2,20,0,0,24,0,60,10,0,150,10,0,180,6  
8,0,300,70,0,340,48,0,362,20,0,362,-2,0,346,-20,  
0,316,-20,0,300,-10,0,296,-2,0,286,-4,0,274,-28,  
0,262,-34,0,270,-32,0,284,-40,0,286,-90,0,200,-9  
4,0,166,-20,0,0,20,0,0  
27020 DATA .4,0,0,10,10,0,40,-4,0,140,-4,0,170,1  
0,0,220,14,0,248,0,0,264,-20,0,262,-52,0,238,-76  
0,0,188,-78,0,162,-30,0,40,-10,0,10,0,10  
27030 DATA .4,20,0,284,40,0,286,50,0,300,52,0,31  
6,40,0,334,26,0,338,12,0,338,0,0,332,-6,0,322,-8  
0,310,-2,0,296,20,0,284  
27040 DATA .7,12,0,346,14,0,342,20,0,340,36,0,34  
2,48,0,334,54,0,324,58,0,326,68,0,332,54,0,350,4  
0,0,356,26,0,356,12,0,346  
27050 DATA .7,-8,0,126,-8,0,134,-10,0,144,-8,0,1  
52,-12,0,158,-20,0,158,-26,0,152,-28,0,142,-22,0  
128,-12,0,124,-8,0,126  
27060 DATA .1,20,0,248,28,0,244,42,0,260,44,0,27  
0,34,0,276,20,0,274,14,0,270,14,0,264,18,0,260,2  
0,0,248  
27070 DATA .6,30,0,309.5,30,36,0,309.64,30,5,0,3  
10,30,36,0,310.36,30,0,310.5,29.64,0,310.36,29.5  
0,310,29.64,0,309.64,30,0,309.5  
27080 DATA .6,30,0,310,30,5,310,31,4.3,310,30,4,  
310  
27090 DATA .3,-58,0,170,-57,0,170,-57,3,170,-55,  
2,170,-55,3,170,-57,10,170,-58,10,170,-60,3,170,  
-60,2,170,-58,3,170,-58,0,170  
27100 DATA .3,-18,0,160,-17,0,160,-17,3,160,-15,  
2,160,-15,3,160,-17,10,160,-18,10,160,-20,3,160,  
-20,2,160,-18,3,160,-18,0,160  
27110 DATA .5,-30000,0,30000,30000,0,30000,.2,20  
0,362,2000,0,1000  
27120 DATA -.8,-4,-42,0,170,-10,0,220,-50,0,226,  
0,260,10,0,310,40,0,310,-4,0,320,-7,40,0,348,1  
6,0,346,-18,0,142,-10,0,152,-24,0,150  
27130 DATA .3,-57.5,5,170,-17.5,5,160,.1,30,0,26

# 3-Dゴルフ・シミュレーション

## リスト3

```

2,42,0,268,16,0,268,0,0,2000,-6,0,288,-5,0,100,1
0000,-2,50,0,366,80,0,370,100,0,260,-.9
28000 *** HOLE17 DATA ***
28010 DATA .2,40,0,0,80,0,90,80,0,140,40,0,194,0
,0,200,-40,0,180,-42,0,172,-36,0,170,-20,0,178,-
10,0,176,-14,0,170,-34,0,156,-36,0,138,-20,0,114
,50,0,110,58,0,106,56,0,100,0,0,96,-36,0,80,-38,
0,56,-30,0,0,40,0,0
28020 DATA .4,10,0,10,20,0,30,24,0,60,8,0,70,-10
,0,70,-22,0,60,-20,0,40,-10,0,10,10,0,10
28030 DATA .4,12,0,118,26,0,126,34,0,140,32,0,15
2,20,0,164,-2,0,166,-20,0,160,-28,0,140,-24,0,12
8,-10,0,120,12,0,118
28040 DATA .7,40,0,120,52,0,118,62,0,130,56,0,15
2,40,0,170,36,0,168,40,0,150,38,0,134,40,0,120
28050 DATA .1,20,0,172,28,0,170,32,0,176,20,0,18
4,4,0,186,-6,0,178,-2,0,172,20,0,172
28060 DATA .6,-10,0,139.5,-9.64,0,139.64,-9.5,0,
140,-9.64,0,140.36,-10,0,140.5,-10.36,0,140.36,-
10.5,0,140,-10.36,0,139.64,-10,0,139.5
28070 DATA .6,-10,0,140,-10,5,140,-11,4.3,140,-1
0,4,140
28080 DATA .3,-38,0,180,-37,0,180,-37,3,180,-35,
2,180,-35,3,180,-37,10,180,-38,10,180,-40,3,180,
-40,2,180,-38,3,180,-38,0,180
28090 DATA .3,2,0,190,3,0,190,3,3,190,5,2,190,5,
3,190,3,10,190,2,10,190,0,3,190,0,2,190,2,3,190,
2,0,190
28100 DATA .5,-10000,0,10000,10000,0,10000,.3,-8
000,0,10000,0,800,10000,5000,0,10000,.2,0,0,200,
-30000,0,5000
28110 DATA -.8,-4,0,0,50,20,0,60,-20,0,60,10,0,1
50,28,0,152,-20,0,140,.7,50,0,140,54,0,150,38,0,
167,.5,-7000,200,10000,5000,200,10000,.3,0,200,1
0000,-2,0,0,1000,.1,-60,0,180,-14,0,174,54,0,104
,-10,0,200,.3,2.5,5,190,-37.5,5,180
28120 DATA .1,12,0,178,28,0,176,.3,2.5,8.5,190,2
5,3.5,190,-.9
29000 *** HOLE18 DATA***
29010 DATA .2,30,0,0,116,130,100,0,260,72,0,34

```

```

0,10,0,450,-68,0,476,-110,0,440,-106,0,386,-70,0
,330,4,0,240,14,0,130,-30,0,0,30,0,0
29020 DATA .4,20,0,10,100,0,124,104,0,176,92,0,2
50,70,0,320,26,0,380,-12,0,388,-26,0,374,-30,0,3
46,0,0,300,26,0,200,30,0,140,0,0,60,-6,0,14,6,0,
6,20,0,10
29030 DATA .4,-50,0,400,-30,0,414,-24,0,430,-30,
0,450,-48,0,460,-60,0,462,-80,0,452,-90,0,430,-
6,0,410,-70,0,400,-50,0,400
29040 DATA .7,10,0,418,12,0,432,0,0,450,-20,0,44
6,-18,0,420,-30,0,400,-24,0,392,-2,0,400,10,0,41
8
29050 DATA .7,60,0,294,66,0,304,54,0,330,36,0,33
6,28,0,332,30,0,320,40,0,304,60,0,294
29060 DATA .1,-38,0,340,-40,0,360,-36,0,380,-40,
0,388,-60,0,390,-80,0,400,-90,0,396,-94,0,380,-8
8,0,360,-68,0,338,-42,0,336,-38,0,340
29070 DATA .1,46,0,124,54,0,128,60,0,160,56,0,19
4,40,0,214,30,212,30,0,200,38,0,170,42,0,130,4
6,0,124
29080 DATA .6,-60,0,429.5,-59.64,0,429.64,-59.5,
0,430,-59.64,0,430.36,-60,0,430.5,-60.36,0,430.3
6,-60.5,0,430,-60.36,0,429.64,-60,0,429.5
29090 DATA .6,-60,0,430,-60,5,430,-59,4.3,430,-6
0,4,430
29100 DATA .3,-18,0,280,-17,0,280,-17,3,280,-15,
2,280,-15,3,280,-17,10,280,-18,10,280,-20,3,280,
-20,2,280,-18,3,280,-18,0,280
29110 DATA .5,-30000,0,30000,30000,0,30000
29120 DATA .7,210,10,200,210,10,400,130,10,400,1
30,10,200,210,10,200,220,0,190,120,0,190,120,0,4
10,130,10,400,130,10,200,120,0,190
29130 DATA -.8,-4,80,0,140,60,0,280,26,0,376,-12
,0,385,-40,0,430,-76,0,430,-34,0,450,-80,0,448,
.7,46,0,316,52,0,328,31,0,331,-6,0,426,0,0,447,-1
8,0,444
29140 DATA .1,-66,0,370,-40,0,382,-88,0,394,48,0
,170,46,0,204,36,0,210,.3,-17.5,5,280,.5,0,500,3
0000,-2,0,0,800,.5,125,5,390,135,5,195,.7,140,10
,390,150,10,360,-.9

```

## R A N D O M B O X

### コンティニューコマンドの使い方

F M-8ディスプレイ・サブシステムの説明のなかったContinueコマンドの使い方を説明します (F M-7のシステム仕様書に書かれています)。

#### コマンド&パラメータ

相 対 値	記 号	内 容	値
0	-		Don't care
1	S'	継続フラグ・ビット(MSB)	S'=1の時、データが継続することを示す。
2	C	コマンド・コード	\$64
3	N	今回転送するバイト数	1~120
4~(3+N)	DATA	転送データ	転送データ

このコマンドは転送するデータ数が多く、一回で転送しきれない場合に使います。初めて送るコマンドの継続フラグ・ビットS'を1にしておきます。

コンティニューコマンドもS'を1にすることでどんどんデータ転送が可能です。

### 拡張BASIC使用上の注意

たとえば、"F-BASIC EXPANDER" の拡張BASICを使う場合、必ず拡張したBASICでテキストを作ってください。こうしないと中間言語が決定しないために、後から拡張しても使えません。

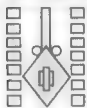
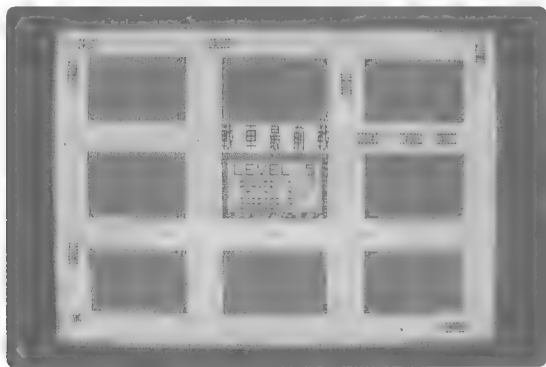


# FM-8

(サブCPU版高速ゲーム)

# 戦車最前線

■高畑英樹



## プログラムの説明

戦車の動きにリアル感を出すため、縦方向に1ドット、横方向に3ドットの移動とし、移動中はキャタヒラの回転を表現しています。表示などに関するプログラムが1Kバイト強サブシステムへ送り込まれますが、ブロック転送のため1秒以内で転送を終了します。

また、ゲーム中でも[STOP]キーが押されると、いつでもBASICのモニタに復帰しReady状態になります。ただし、CONT命令は使えません。その他の命令は表示も含めてすべてOKです。

音出力は、サブからメインへの高速割り込み(FIRQ)で処理しています。ただし、低音域のノイズの減衰のため、本体内部のスピーカーでは低域の音響処理がまったくなされていないので、どうしてもあのズードドドーンとかドドドーンという音が出ません。そこで必ず外部スピーカーを必要とします。カセット・インターフェイスのリード線(MIC)に出力しています。高音は下げた方がよいかもしれません。

デジタルで表現される減衰音は、何か物を引き裂くような音を伴った独特の音で、結果的に味のあるバクハツ音が得られました。許される範囲で音量を上げてプレイすると、相手をやっつけたときの快感は一段と増し、ストレス解消にもなります。

プログラムのマシン語エリアは\$5000~\$658Eです。ゲーム・スタートはBASICからRUNしてください。



## ゲームの説明

全部で5面のゲームで、5面全部クリアするとあなたが陸軍大佐に任命される。という一種のシュミレーション的ゲームになっています(深く追求しない)。

各面クリアごとにレベルが上がります。最初は2台の戦車ですが最大レベル5では10台の戦車を相手に戦います。

各面を1分以内でクリアした場合、その時間に応じて次の式で計算した数のバリアが与えられます(7-秒/10)。ただし最大9個までです。

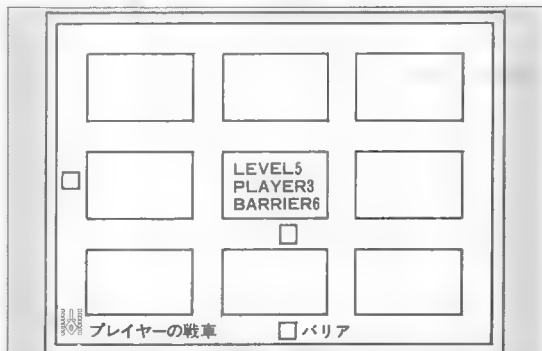
高レベルにそなえて低レベルで早くクリアし多くのバリアを獲得しておきましょう。

相手との対戦では必ず相手の後方より攻撃して、すばやくその場所から立ち去ります。向かいあつての撃ちあいは、必ずあなたが勝つとは限りません。

バリアは原則としてどここの場所でも張れますが、交差点付近では戦車に対して効きめのないときがありますから注意してください。しかし、弾丸に対してはすべてのところで効きめがあります。

また、バリアはあなたの戦車が通ると壊れます。バリアに近づいて弾丸を発射しても壊れますが、1キャラクタ分以上間があると壊れません。なお速度はプレイヤーの方が相手の倍速になっています。キー操作はTV画面を見てください。

図1 バリアの張り方の良い例



注) レベル4か5に入った直後、この3箇所にバリアが張れるとクリア間違いなし。



## 最後に

完全なオリジナル・ゲームのため、最初はレベル2~3

ぐらいでとまどいますが、少しやって目が慣れてくると、レベル3~4ぐらいまではバリアなしでも勝てるようになります。問題はレベル4~5にあるようです。

なお、カセット・サービスではFM-7/8用が一本になっています。

### BASICプログラム・リスト

```

10 COLOR7,0:WIDTH80,20:DIMA%(15):SYMBOL(160,10),"*** GAME Message ***",2,2,7
20 LOCATE0,3:PRINT" センシャ ノ ソウサ ハ カール イトウキー ラ オシテ ソレゾレ ノ ホーコー ヲ イトウ サセマス。"
   ":PRINT:PRINT" テイシ ハ スグー スキー。":PRINT:PRINT" コウグキ ハ テイシ サセテカラ スグー スキー ラ
   オス。":PRINT:PRINT" ハリヤ ラ センシャ ノ マエ ニ ハル トキハ ソノタ ノ キー ラ オシマス。"
30 PRINT:PRINT"テキ ノ センシャ ラ コウグキ スルトキ ハ カナラス" テキ ノ センシャ ノ ウシロ ヲ マワリ センシャ カ フリカエ
   ヂテ アタラ コウグキ スルマエニ テイシ シテ コウグキ シタ アト スハ" タク ソノ ハ" ショカラ タチサツテ クダ" サイ。 オダカ"イ ニ
   ムカイアツテ ノ コウグキ ハ アタラ カ" カナラス" カツ トハ" カキ" リマセン。":PRINT
40 PRINT"ハ" リヤ ハ コウサテン ノ シューエン ハ テキ ノ センシャ ニ タイシテ キキメ ノ ナイトキ カ" シハ"シハ" アリマス。タダ"シ
   コウサテン ノ ナカ ハ ソノタ ノ トコロ ト"ウヨウ ニ コウリョク カ" アリマス。タン"カン"ニ タイスル キキメ ハ スグー"テノ トコロ
   テ" コウリョク カ" アリマス。ソレゾレノ レベル ラ イッパ"ン イナイ テ" クリヤー スルト シ"カン"ニ オオシ"テ ハ"リヤ カ" ツイカ
   サレマス。(9コ マテ)"
50 I$=INPUT$(1):WIDTH80,20:-----CON WIDTH80,20 ハ フ"ン 10 ニ カカワラス" カナラス" イル ノダ"!!
60 LINE(0,0)-(639,199),PSET,4,BF:FOR Y=0 TO 639 STEP 2:LINE(Y,0)-(Y,199),PSET,2:NEXT
70 GOSUB 220:GOSUB 230:GOSUB 240:SYMBOL(263,87),"LEVEL 5",2,1,2:POKE&H6526,10:SYMBOL
   L(272,97),"PLAYER 3",1,1,0:SYMBOL(272,107),"BARRIER 9",1,1,0:POKE&H6527,255:EXEC
   C&H5000:I=PEEK(&H6528):IFI=3 THEN GOSUB 220:GOSUB 230:GOTO 80 ELSE 70
80 FMSUB=2:LEVEL=1:PLAYER=3:BARR=0:SYMBOL(160,122),"GAME START",4,2,3:FOR I=0 TO 20
   00:NEXT:GOSUB 220:TIME$="00:00:00":GOTO 170
90 GOSUB 220
100 IFFM=1 THEN LEVEL=LEVEL+1:FMSUB=FMSUB+2:GOTO 120
110 IFFM=2 THEN PLAYER=PLAYER-1:GOTO 160
120 SYMBOL(360,184),"TIME "+TIME$,2,1,0:FOR I=0 TO 20000:NEXT:GOSUB 220
130 IF ATIME>0 THEN 150 ELSE I=7-INT(BTIME/10):IFI<0 THEN I=0
140 BARR=BARR+1:IF BARR>9 THEN BARR=9
150 IF LEVEL=6 THEN GOSUB 220:SYMBOL(8,122),"CON ジョミレータ ハ アタラ リクワ"ン タイサ ニ ニンメイ シマス。
   ",2,2,1:LOCATE0,0:END
160 GOSUB 230
170 I$=STR$(LEVEL):SYMBOL(263,87),"LEVEL"+I$,2,1,2:POKE&H6526,FMSUB
180 I$=STR$(PLAYER):SYMBOL(272,97),"PLAYER "+I$,1,1,0
190 I$=STR$(BARR):SYMBOL(272,107),"BARRIER"+I$,1,1,0:POKE&H6527,BARR
200 IF PLAYER=0 THEN SYMBO(172,122),"GAME OVER",4,2,2:FOR I=0 TO 5000:NEXT:GOTO 70
210 EXEC&H5000:BARR=PEEK(&H6528):FM=PEEK(&H6528):ATIME=PEEK(&H6529):BTIME=PEEK(&
   H6529):GOTO 90
220 FOR Y=4 TO 120 STEP 58:LINE(8,Y)-(631,Y+17),PSET,5,BF:LINE(8,Y+18)-(55,Y+57),PSET
   ,5,BF:LINE(200,Y+18)-(247,Y+57),PSET,5,BF:LINE(392,Y+18)-(439,Y+57),PSET,5,BF:LI
   NE(584,Y+18)-(631,Y+57),PSET,5,BF:NEXT Y:LINE(8,178)-(631,195),PSET,5,BF:RETURN
230 LINE(258,85)-(381,114),PSET,6,BF:RETURN
240 SYMBOL(165,122),"DEMONSTRATION",3,2,7
250 RESTORE 260:FOR I=248 TO 376 STEP 32:FOR J=0 TO 15:READ A%(J):NEXT J:PUT$(I+1,63)-(I+16
   ,78),A%,PSET,0:PUT$(I,63)-(I+15,78),A%,PSET,0:NEXT I:RETURN
260 DATA-28128,-28120,-27612,32,-466,-27664,-28128,-478,-28126,-28126,-492,4116,
   4104,-488,4196,4482
270 DATA 256,-2,256,256,32764,16644,16644,32764,16644,16644,32764,256,256,-2,256,
   256
280 DATA 0,16376,8200,16376,8200,16376,0,-2,8704,16124,8708,15940,8744,11792,-354
   4,710
290 DATA 2080,1088,-2,0,4,32324,16964,16964,32324,16964,16964,32324,16964,16900,1
   6900,17932
300 DATA-28128,-28120,-27612,32,-466,-27664,-28128,-478,-28126,-28126,-492,4116,
   4104,-488,4196,4482

```

### マシン語ダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5000	B6	FD	05	2B	FB	1A	50	86	80	B7	FD	05	B7	FD	05	CE	:BE
5010	FC	80	30	8D	00	96	C6	4A	A6	A0	A7	C0	5A	26	F9	7F	:64
5020	FD	05	7F	FD	05	8E	00	C8	30	1F	26	CE	62	75	8E	:7D	
5030	D1	00	86	09	C6	31	8D	33	7F	FD	05	7F	FD	05	CE	:64	:48
5040	29	8E	7E	80	86	08	C6	30	8D	21	7F	FD	05	7F	FD	:05	:E9
5050	33	8D	01	DF	8E	8E	80	86	54	C6	34	8D	0E	B6	01	:B7	:19
5060	FC	80	7F	FD	05	7F	FD	05	16	00	89	DD	00	B6	FD	:05	:B2
5070	2B	FB	D7	01	86	B0	A7	0A	05	B7	FD	05	BF	FC	FE	:3A	:6F
5080	10	8E	FC	CA	A6	C0	A7	0A	01	26	FB	F7	FC	B0	7F	:2C	
5090	FD	05	7F	FD	05	86	FD	05	2B	FB	86	B0	B7	FD	05	:B7	:D7
50A0	FD	05	B6	FC	80	26	E8	0A	00	26	C7	39	00	00	3F	:59	:0A
50B0	41	4D	41	55	43	48	49	93	D3	BF	90	B7	D4	0A	B6	:D4	:9C
50C0	08	B6	D4	09	B6	D4	0A	F7	D3	80	7F	D3	80	7F	D3	:80	:A5
50D0	B6	D4	0A	B6	D3	FE	20	F8	B1	01	27	15	B7	D4	0A	:B7	:C6
50E0	D4	0A	FE	D3	FE	8E	D3	CA	E6	80	E7	C0	4A	26	F9	:20	:6E
50F0	D6	7E	BE	80	30	8D	00	07	BF	FF	F6	1C	00	20	FE	:34	:78

Sum	B6	0F	1B	45	8A	B7	76	0D	D2	A2	8E	D8	B1	DD	8B	28	:D1
5100	76	B6	FD	04	84	02	26	17	8E	01	E0	BF	FF	F6	86	:40	:D9
5110	B7	FD	05	B7	FD	05	30	8D	FF	91	AF	69	35	76	38	:86	:43
5120	B0	B7	FD	05	B7	FD	05	87	FD	05	FC	FE	10	BE	FC	:6B	
5130	F0	7F	FD	05	7F	FD	05	C1	FF	27	CD	C1	00	26	02	:00	:AF
5140	DB	C1	04	24	09	FD	65	27	10	BF	65	29	20	CE	C0	:05	:66
5150	26	06	30	8D	00	14	20	C2	5A	26	06	30	8D	00	5B	:20	:9D
5160	B9	5A	26	B8	30	8D	00	B8	20	B0	86	C8	97	00	BE	:A0	:1C
5170	00	A6	B0	2B	10	7F	FD	00	31	3F	31	3F	48	26	F4	:0A	:29
5180	00	26	EE	20	07	CE	61	F7	FD	00	20	EC	86	64	97	:00	:C3
5190	A6	B0	2B	17	7F	FD	00	D6	00	53	0A	00	26	03	16	:FF	:55
51A0	5C	5A	12	26	FB	FB	26	E9	20	E5	C6	41	F7	FD	00	:52	
51B0	D6	00	5B	0A	00	26	EA	16	FF	43	CC	01	2C	DD	00	:DD	:53
51C0	02	BE	A0	00	A6	B0	2B	18	F7	FD	00	97	05	DC	00	:C3	:50
51D0	00	02	DD	00	83	00	01	26	FB	96	05	48	26	EB	20	:E4	:79
51E0	C6	41	F7	FD	00	97	05	DC	02	83	00	02	DD	02	26	:E4	:E3

51F0	16	FF	0A	CC	00	3C	DD	00	DD	02	7F	FD	00	DC	00	C3	FE	
Sum:	0D	80	D7	75	D5	55	63	B3	B2	60	D9	D6	DF	73	0E	DB	E5	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
5200	01	DD	00	83	00	01	26	FB	0C	02	83	00	01	DD	02	C4		
5210	26	03	16	FE	EB	BE	A0	00	97	05	A6	80	84	01	26	0C	CC	
5220	7F	DD	00	96	05	83	00	01	26	EE	20	CE	B6	41	B7	FD	1B	
5230	00	F0	20	F0	24	BE	C0	00	6F	80	BC	07	D0	26	F9	B6	59	
5240	07	A7	80	BC	CF	0A	26	F9	B6	FF	B7	D3	FF	B4	0A	04	E4	
5250	B6	04	0A	B6	04	0A	26	F9	B6	FF	B7	D3	FF	B4	0A	04	FB	
5260	BD	FF	D2	BF	D0	00	B6	D4	02	1C	00	16	0E	10	A6	A4	13	
5270	C6	50	30	1F	01	E6	21	3A	A6	A4	43	10	27	03	39	A6	5A	
5280	22	E6	24	10	27	00	58	5A	27	03	16	03	2B	E6	25	6C	FA	
5290	25	C0	0F	27	14	A6	22	4A	10	27	06	91	4A	10	27	06	96	
52A0	46	4A	10	27	06	B0	16	05	FE	6F	24	6F	25	A6	22	4A	9C	
52B0	26	07	C6	03	E7	22	16	06	6D	4A	26	09	C6	04	E7	22	D4	
52C0	BD	16	05	DF	0A	26	07	C6	01	E7	22	16	06	5E	C6	24		
52D0	02	E7	22	BD	03	16	06	0F	34	10	17	06	7D	35	90	4A	B3	
52E0	10	26	01	60	EC	B9	20	D6	F6	C3	00	10	26	02	C5	EC	2F	
52F0	A4	CB	05	D1	4F	26	D0	D6	50	90	4E	24	14	43	80	23	09	
Sum:	AR	00	BD	FB	4D	80	5B	9F	2B	49	D5	03	F2	26	FB	0C	5C	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
5300	10	25	FF	91	C1	04	26	1C	17	02	CB	48	27	32	16	FF	166	
5310	84	80	2F	10	25	FF	7E	C1	02	26	09	17	02	BB	48	27	117	
5320	69	16	FF	71	A6	A4	D6	4E	B1	0A	26	44	C1	11	10	25	59	
5330	00	E1	A6	21	B1	16	26	00	6F	15	B4	01	10	26	00	C1	B9	
5340	A6	B9	85	A5	43	10	26	02	6B	A6	B9	88	C5	43	10	26	34	
5350	02	62	17	06	1F	30	26	00	D0	6C	22	C6	05	3A	EB	21	A5	
5360	E7	21	16	05	B2	B1	2E	27	DF	B1	46	27	D3	16	00	91	B2	
5370	B1	46	26	53	C1	4B	24	06	C1	32	10	24	00	95	A6	21	F7	
5380	B1	16	26	32	96	0C	B4	01	26	A6	B9	7C	EC	43	10	26	CD	
5390	26	02	21	A6	B9	79	C5	43	10	26	02	18	17	05	D5	B6	C0	
53A0	04	A7	22	30	B9	7A	65	A6	A4	80	0B	A7	A4	A6	21	B8	D7	
53B0	05	A7	21	16	0A	EE	B1	2E	27	CA	B1	46	26	43	96	15	B0	
53C0	B4	01	26	C6	16	FF	79	B1	7E	26	18	C1	B5	24	04	C1	168	
53D0	6C	24	A0	A6	21	B1	16	27	A7	B1	2E	27	D9	B1	46	27	99	
53E0	D9	20	1E	B1	B8	26	1A	C1	A6	24	28	A6	21	B1	16	26	07	
53F0	08	96	15	B4	01	26	0A	20	91	B1	2E	27	F4	B1	46	27	D1	
Sum:	BE	2D	CE	C5	4E	B2	B3	A5	5B	76	4F	B0	67	C3	B4	70	101	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
5400	B9	A6	23	BB	54	B1	FC	26	05	4F	30	01	6C	21	A7	23	B0	
5410	16	05	1A	97	32	96	4F	A0	21	10	25	01	B8	B6	0A	97	1BC	
5420	C3	0C	D0	59	EC	C4	33	49	31	32	27	00	0A	33	26	F4	A4	
5430	17	07	91	48	27	CB	16	05	66	E1	21	23	EF	D1	4F	24	BC	
5440	EB	16	FE	51	4A	10	26	01	A6	EC	B9	86	0E	C3	00	01	116	
5450	10	26	01	60	EC	A4	B9	0E	91	4E	26	2D	96	50	D0	4F	F7	
5460	24	14	53	C0	08	10	25	FE	2C	B1	03	26	1C	17	01	66	F9	
5470	48	27	32	16	FE	1F	C0	10	01	25	FE	19	B1	01	26	09	A1	
5480	17	03	53	48	27	32	16	FE	0C	96	4F	E6	21	C1	03	26	39	
5490	44	B1	05	10	25	00	0A	A6	A4	B1	36	26	2C	96	15	B4	5B	
54A0	01	10	26	00	C4	A6	B9	85	A4	10	26	01	06	A6	B9	B2		
54B0	A6	85	58	43	10	26	00	AD	17	04	A0	30	89	7E	C0	A6	22	91
54C0	A6	A4	B8	0E	A7	A4	16	04	A4	B1	70	27	D0	B1	AA	10	1CF	
54D0	26	00	96	20	D0	C1	1B	26	E2	B1	10	24	06	B1	16	10	16F	
54E0	24	00	8E	A6	A4	B1	36	26	30	96	0C	84	01	26	AE	A6	1AA	
54F0	B9	85	4D	43	10	26	00	BC	A6	B9	85	49	43	10	26	00	106	
Sum:	AA	04	DF	C9	39	A4	07	7D	24	6D	30	F7	E9	2B	D9	AC	10B	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
5500	B3	17	04	56	30	B9	7E	BC	6C	22	A6	A4	B8	0E	A7	A4	1D3	
5510	A6	21	80	0A	A7	21	16	04	00	10	27	0C	B1	A4	26	16F		
5520	48	96	15	B4	01	10	26	FF	7C	D0	C4	C1	33	26	1A	B1	C2	
5530	35	24	06	B1	2E	10	24	00	3B	A6	A4	B1	36	27	A4	B1	CD	
5540	70	27	A6	B1	A4	27	DA	20	C0	C1	4B	26	1C	B1	46	24	E2	
5550	20	A6	A4	B1	36	26	08	96	15	B4	01	26	92	20	A4	B1	E2	
5560	70	27	F4	B1	A4	27	BB	20	00	6C	A4	30	B8	50	16	03	B6	
5570	76	D7	32	96	4E	A0	A4	10	25	00	2D	B6	0A	97	33	CE	31	
5580	D0	59	EC	A1	D1	32	27	0F	33	49	0A	33	26	F4	17	00	1FC	
5590	33	28	27	D5	16	04	08	A1	A4	23	ED	91	4E	24	E9	A6	B0	
55A0	42	B4	01	26	E3	16	0C	ED	A6	9C	0C	B4	01	27	06	96	57	10
55B0	10	27	FC	E1	B6	01	A7	24	3A	20	10	8E	00	64	31	3F	2C	
55C0	26	FC	E1	B6	01	A7	24	3A	20	10	8E	00	64	31	3F	2C	DE	
55D0	39	B6	02	A7	24	39	34	10	BE	D0	5C	0C	A4	05	97	32	16C	
55E0	A6	B4	B1	02	27	07	3A	0A	32	26	F5	96	52	35	90	4A	163	
55F0	10	26	01	4E	EC	B9	80	EE	C3	00	01	26	B7	EC	A4	D1	16A	
Sum:	B6	35	DB	AF	FB	46	2D	70	D2	AF	0F	F6	32	11	66	CC	14B	
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum	
5600	4F	20	D6	50	90	4E	24	14	43	80	23	10	25	FC	85	F5	7A	
5610	C1	04	26	1C	17	FF	BF	48	27	32	16	FC	78	80	2F	10	1C6	
5620	25	FC	72	C1	02	26	09	17	FF	AC	48	27	62	16	FC	65	18B	
5630	A6	A4	D6	4E	B1	0A	26	3D	C1	11	10	25	00	D3	A6	21	1FD	
5640	B1	33	26	96	15	B4	01	10	26	00	83	B6	B9	85	A0	16C		
5650	43	10	26	FF	5F	A6	B9	88	C0	43	10	26	FF	5F	A6	B9	88	157
5660	13	30	89	7D	D0	6A	22	16	02	7D	B1	1F	27	D6	B1	03	157	
5670	27	DA	16	00	B8	B1	44	26	A4	C1	4B	24	06	C1	32	10	10F	
5680	24	00	8E	A6	21	B1	33	26	2A	96	0C	B4	01	26	B5	A6	125	
5690	B9	7C	0E	43	10	26	FF	1C	A6	B9	79	C0	43	10	26	FF	1C	15D
56A0	13	17	02	D0	30	B9	7A	60	6C	22	A6	B4	00	80	B7	A4	139	
56B0	16	01	F1	B1	1B	27	D2	B1	03	26	46	96	15	B4	01	26	1E1	
56C0	CE	20	B9	B1	7E	26	18	C1	B5	24	04	C1	6C	24	A2	A6	15B	
56D0	21	B1	33	27	B4	B1	1B	27	B0	B1	03	27	DE	20	20	B1	16D	
56E0	BB	26	1C	C1	A6	24	2A	1B	A6	21	B1	33	26	08	96	15	B4	187
56F0	01	26	9C	20	0A	B1	1B	27	F4	B1	03	27	92	20	00	A6	1A7	
Sum:	57	98	5B	66	97	0B	A5	5D	A0	E7	76	36	79	C3	16	91	167	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5700	23	BB	54	B1	FC	26	05	4F	30	1F	6A	21	A7	23	16	02	1B5
5710	15	97	32	96	4F	A0	21	10	24	FE	BD	86	0A	97	33	CE	16B
5720	D0	59	EC	C4	33	49	91	32	27	00	0A	33	26	F4	17	FE	1B8
5730	93	48	27	CB	16	02	68	E1	21	24	EF	D1	4F	23	EB	16	1A

# 戦車最前線

## マシン語ダンプ・リスト

5C30	F6	D4	04	39	86	07	EF	64	3A	4A	26	FA	39	96	57	26	:B7
5C40	04	0C	57	20	94	86	02	91	52	26	02	20	BC	97	52	86	:C9
5C50	05	8D	04	16	FF	83	96	58	10	27	02	76	96	50	4A	10	:D8
5C60	26	00	AD	DC	4E	81	0A	26	2A	C1	1B	26	16	96	58	4A	:25
5C70	27	75	4A	26	72	0C	50	17	03	97	96	4F	8B	03	97	4F	:E4
5C80	16	02	4F	C1	30	27	E6	C1	48	26	C5	96	58	4A	4A	27	:99
5C90	E4	20	54	B1	44	26	2B	C1	1B	26	1A	96	58	B1	04	26	:1D
5D00	CC	97	50	17	03	6B	96	4E	8B	0B	97	4E	96	4F	8B	03	:1F
5D10	97	2F	16	02	1D	C1	30	27	E2	C1	48	27	DE	20	28	B1	:EC
5D20	7E	46	0E	C1	1B	27	D4	C1	30	27	D0	C1	48	27	CC	20	:8A
5D30	16	C1	1B	26	08	96	58	B1	04	27	C6	20	0A	C1	30	27	:BF
5D40	F4	C1	48	27	F0	20	00	96	58	B1	03	26	0B	97	50	17	:D2
5D50	03	1F	16	01	DD	D6	4F	C1	48	10	27	01	D5	96	51	8B	:C3

Sum: 42 DB 10 DA 91 2B C1 A0 39 72 9A 52 6D EA 60 A5 :1E

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5D00	54	97	51	B1	FC	10	26	01	C9	0F	51	0C	4F	16	01	C2	:4D
5D10	4A	10	26	00	9B	DC	4E	C1	03	26	26	B1	39	26	13	96	:DB
5D20	58	4A	26	71	17	03	02	0A	50	96	4E	8B	0B	97	4E	16	:24
5D30	01	A0	B1	73	27	E9	B1	AD	26	5B	96	58	4A	27	E5	20	:8B
5D40	54	C1	1B	26	2B	B1	39	26	1A	96	58	B1	03	26	D0	17	:F7
5D50	02	D7	0C	50	96	4E	8B	0B	97	4E	96	4F	80	04	97	4F	:E3
5D60	16	01	6F	B1	73	27	E2	B1	AD	27	DE	20	28	C1	33	26	:18
5D70	0E	B1	39	27	D4	B1	73	27	D0	B1	AD	27	CC	20	16	B1	:8A
5D80	39	26	08	96	58	B1	03	27	C6	20	0A	B1	73	27	F4	B1	:80
5D90	AD	26	02	20	EE	96	58	B1	04	26	05	97	50	16	01	32	:B1
5DA0	96	4E	B1	AD	10	27	01	2A	0C	4E	16	01	25	4A	10	26	:8A
5DB0	00	9B	DC	4E	B1	0A	26	1C	C1	32	26	10	96	58	B1	02	:2C
5DC0	26	02	17	02	4C	0A	50	0C	4F	16	01	06	C1	1A	27	EC	:AD
5DD0	C1	02	27	E8	B1	4A	26	2A	C1	32	26	16	96	58	B1	04	:83
5DE0	26	D4	17	02	2C	0C	50	0C	4F	96	4E	80	0B	97	4E	16	:66
5DF0	00	E0	C1	1A	27	E6	C1	02	27	E2	20	28	B1	7E	26	0E	:0F

Sum: FA FE 6A 3A CE D7 19 7E BD 3B B4 74 B5 6B 99 BA :0B

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5E00	C1	32	27	DB	C1	1A	27	D4	C1	02	27	D0	20	16	C1	32	:AB
5E10	26	08	96	58	B1	04	27	CA	20	0A	C1	1A	27	F4	C1	02	:75
5E20	26	02	20	EE	96	58	B1	01	26	0B	97	50	17	01	E2	16	:CB
5E30	00	A0	D6	4F	C1	02	10	27	00	9B	96	51	8B	54	97	51	:05
5E40	B1	FC	10	26	00	8C	0F	51	0A	4F	16	00	85	DC	4E	C1	:7E
5E50	03	26	10	B1	7E	26	0F	96	58	4A	26	62	17	01	CA	B6	:A2
5E60	01	97	50	16	00	6C	B1	44	27	DE	B1	0A	27	E9	20	4E	:4C
5E70	C1	1B	26	02	B1	7E	26	1A	96	58	B1	03	26	D9	17	01	:B6
5E80	AB	0A	50	96	4F	80	04	97	4F	16	00	46	B1	44	27	E8	:B1
5E90	81	0A	27	E4	20	2B	C1	33	26	0E	B1	7E	27	DA	B1	4A	:CR
5EA0	27	D6	B1	0A	27	D2	20	16	B1	7E	26	08	96	58	B1	03	:56
5EB0	26	0C	27	CA	B1	44	27	F4	B1	0A	27	F0	20	00	96	58	:83
5EC0	B1	02	26	04	97	50	20	0A	96	4E	B1	0A	27	04	0A	4E	:80
5ED0	20	00	96	4E	C6	50	3D	1F	01	D6	4F	3A	10	8E	D0	4E	:92
5EE0	96	50	4A	27	0B	4A	27	67	04	10	27	00	8C	16	00	EB	:45
5EF0	A6	05	84	0F	B1	0F	27	67	04	B1	0A	26	7A	AB	B9	01	:E5

Sum: A6 FD FF 22 9B CR 5B 6D FF 74 3E 74 99 A5 E4 21 :57

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5F00	B4	0F	B1	0F	27	04	B1	26	AC	17	FA	D0	20	30	B9	BD	:12
5F10	D0	CE	FF	FF	96	51	26	12	CC	0F	B8	BD	04	30	B9	01	:D9
5F20	E0	A7	B4	EF	01	EF	02	E7	04	39	28	0F	86	01	BD	04	:62
5F30	30	B9	01	E0	A7	B4	EF	01	EF	03	39	CC	3F	E0	BD	04	:5C
5F40	30	B9	01	E0	6F	B4	A7	01	EF	02	EF	03	E7	05	39	A6	:E3
5F50	89	05	50	43	27	04	B1	55	26	1C	A6	B9	05	51	43	27	:53
5F60	04	B1	55	26	11	17	F9	F7	30	B9	BA	60	6F	B4	6F	01	:D6
5F70	30	B8	50	16	00	82	0C	56	39	A6	BA	64	F0	B1	F0	27	:71
5F80	04	B1	A0	26	F1	A6	B9	01	E0	B4	F0	B1	F0	27	04	B1	:DD
5F90	A0	26	E3	17	F9	90	30	B9	BD	D0	CE	FF	FF	96	51	26	:68
5FA0	12	CC	1F	F0	BD	04	30	B9	01	E0	A7	01	EF	02	EF	03	:A3
5FB0	E7	05	39	28	0F	86	B0	8D	04	30	B9	01	E0	EF	01	EF	:6F
5FC0	03	A7	05	39	CC	0F	FC	8D	04	30	B9	01	E0	A7	84	EF	:1C
5FD0	01	EF	02	E7	04	6F	05	39	A6	BA	43	27	04	B1	55	26	:1E
5FE0	95	A6	01	43	27	04	B1	55	26	BC	17	F8	B7	30	B9	38	:EC
5FF0	50	BD	05	6F	B4	6F	01	39	86	0E	C6	50	CE	E0	07	EF	:CC

Sum: D7 E5 E3 66 0D 92 B1 23 5B 86 DD C4 5B 82 B5 93 :4F

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6000	B4	3A	4A	26	FA	39	96	4E	C6	50	3D	1F	01	D6	4F	3A	:17
6010	39	BD	F3	17	F9	5E	30	B9	BD	D0	4F	5F	9D	04	30	B9	:E5
6020	01	E0	ED	B4	ED	02	ED	04	39	BD	BD	17	F9	2C	30	B9	:1B
6030	BA	60	86	12	C6	50	CE	00	00	EF	BA	3A	4A	26	FA	39	:E6
6040	3C	00	FF	F0	FF	FF	FF	00	7B	00	03	B0	1F	E0	1F	F8	:87
6050	03	F0	01	C0	80	01	B0	00	7C	01	FF	FF	FF	0F	FE	C1	:E0
6060	00	3C	01	B0	01	B0	03	C0	0F	F8	1F	1F	E0	03	C0	:1D	
6070	F8	3C	3D	E0	7E	DF	0F	FB	07	FF	1E	70	3F	3C	0F	0B	:DA
6080	0F	0C	8E	D0	4E	86	7B	6F	4A	26	F8	86	0A	C6	03	:7B	
6090	CE	01	00	8E	D0	59	10	8E	00	ED	84	CB	07	EF	02	:62	
60A0	30	09	31	3F	26	F4	86	AD	C6	03	DD	4E	86	04	97	:50	
60B0	97	57	86	08	80	7F	7D	C6	03	BE	D0	59	CE	FF	FF	:A4	
60C0	27	0B	EF	B4	E7	04	30	09	20	F5	0F	04	10	BE	D0	:85	
60D0	10	9F	CB	A6	2B	26	50	A6	24	27	08	4A	27	05	4A	:10	
60E0	27	01	05	17	F1	8B	10	9E	0B	C1	29	10	BC	D0	B3	:26	
60F0	DF	B6	7F	7E	2A	0B	86	03	D6	04	10	26	01	1E	17	:F8	

Sum: 90 36 6E 4A 43 D3 B7 D3 75 45 3C 60 B6 BC 1B 6C :6A

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6100	95	96	56	10	26	00	F8	10	BE	01	F4	31	3F	26	FC	B6	:BA
6110	7F	7E	2A	0B	86	03	D6	04	10	26	01	00	17	F8	77	96	:E5
6120	56	10	26	00	DA	20	A5	A6	28	B1	01	27	10	BD	23	A6	:0B
6130	2B	26	A4	7A	7F	7D	26	9F	86	01	16	00	DF	17	00	9D	:5D
6140	A6	22	B4	01	26	07	17	F8	11	BD	07	20	BA	17	F8	24	:0B
6150	20	F7	A6	2B	6C	28	4A	26	32	86	06	B7	D3	FF	B6	D4	:BA
6160	04	B6	D4	0A	B6	D4	0A	BD	7A	A6	22	B4	01	26	16	30	:DA
6170	89	00	EF	AF	AB	5B	0C	FF	FF	ED	A4	17	00	BF	30	B9	:14
6180	3C	40	16	00	CF	30	89	FF	F1	20	EB	AE	AB	5B	B1	04	:8B
6190	26	07	30	89	B0	00	16	00	A4	B1	07	26	17	30	B9	B0	:1E
61A0	F1	CE	7F	B2	86	06	97	32	EC	C1	ED	B4	30	BB	50	0A	:75
61B0	32	26	F5	39	81	09	27	01	39	6F	2B	EC	FF	FD	10	71	:A1
61C0	30	89	7C	40	BD	0A	AE	A8	5B	30	B9	40	00	CE	00	00	:84
61D0	86	0C	C6	50	EF	8A	EF	02	3A	4A	26	F8	39	A6	A4	C6	:7F
61E0	50	1D	30	1F	01	E6	21	3A	39	17	F7	B4	93	56	26	10	:A6
61F0	24	10	27	FE	F1	17	F7	AF	96	56	26	03	16	FE	E7	96	:A5



# 3次元迷路



市川浩志

Kコンパイラを入手したので、さっそく3次元迷路のプログラムをK用にコーディングし直してみました。その結果、BASICでは迷路を作るのに1分以上もかかっていたのが、なんとKコンパイラではわずか0.7秒ノイライラする待ち時間がなくなって、ゲームが一層楽しくなりました。

## ゲームについて

3次元迷路の大きさはX座標、Y座標、Z座標ともに0から12までです。Z軸の方向を上下方向と見なして、0階から12階までの立体迷路とし、迷路地図は階別に表示します。つまり、Z座標が4ならば4階、10なら10階ということです。地図の中には、自分のいまいる位置が“ロ”で表示されます。また、出口は白で表示されます。

操作するキーは次の通りです。

F, 5  
L, 4  
R, 6  
U, 8  
D, 2  
B, 0  
M, +

前に進む。

左に向きを変える。

右に向きを変える。

上に向きを変える。

下に向きを変える。

後ろに向きを変える。

MAPを見る（このキーを押した後で、何階から何階までの地図を見るのかを入力します）。

RETURN

MAPを見る（このキーでいま自分がいる階の地図が表示されます）。

G

GIVE UP（迷路の脱出をあきらめたときに押してください。このあとで

N以外のキーを押せば再ゲームになります）。

以上のキーを操作して、チャージが0になる前に迷路を脱出してください。チャージは時間の経過と共に減っていきます。地図は何回でも見ることができますが、1つの階を見るごとにチャージが減ります。得点は脱出までにかかった時間と地図を見た回数をもとに計算されますから、あまり多く地図を見ると得点がマイナスになることもありますよ。

画面の下の方には、いまいる位置のX、Y、Z座標が表示されますから、動いたときの座標の変化とMAPとから、

3次元迷路の画面

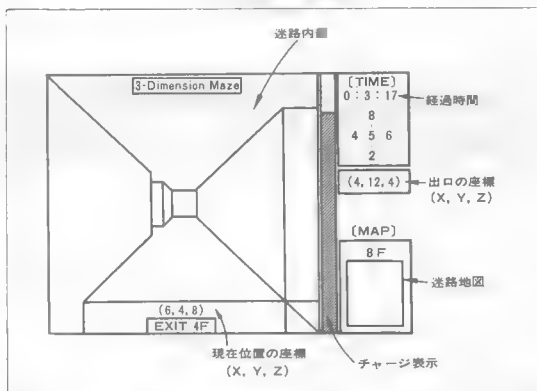


写真1 このまいくと下に落ちるゾ。

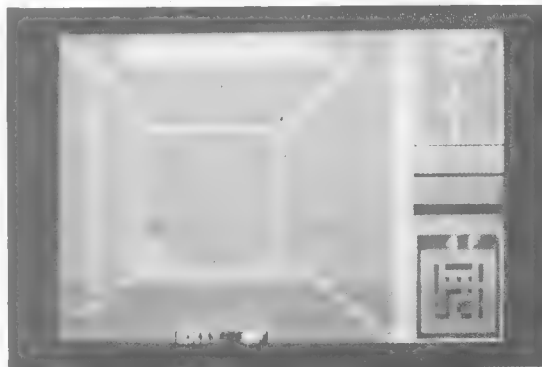
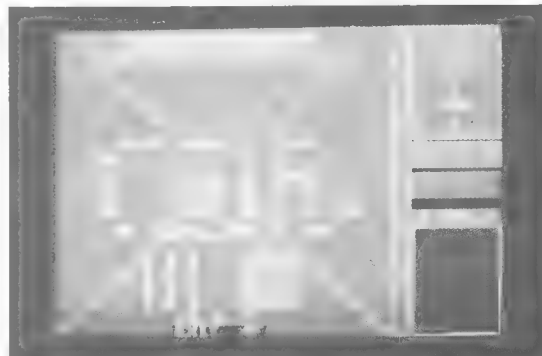


写真2 出口がわからずGIVE UP





# 3次元迷路

自分がどっちを向いているのかをまずつかむことが大切です。

## プログラムについて

Kコンパイラの配列は1次元配列だけですが、関数コールがあるので3次元配列と同じ感覚でプログラムが書けました。

配列には1バイト配列と2バイト配列があるので、入力のとくに気をつけてください。実は、このプログラムを作ったばかりの間バグに悩まされたのですが、その原因はなんと負の数を要素に持つ1バイト配列と単純変数とを加えてしまったことにあったのです。BASICでは考えられないミスですが、おかげでSEX関数の存在理由がわかりました。

ソース・プログラムの40行はF-BASICのRANDOMIZEと同じように乱数の系列を変更するためのものです。

ソース・プログラムの変更は簡単ですから、ゲームに慣れてきたら地図の減点を増やしたり、制限時間を短くするなどしてください。ただし、メモリの余りが少ないのでプログラムの追加はほとんどできません。

## コンパイルについて

ソース・プログラムが長いので、コンパイルするにはファイル版の拡張Kコンパイラが必要ですが、ソース・プログラムのデバッグ用にオン・メモリ版も用意してください。

というのは、カセットテープをファイルとして使う場合、ファイル版のKコンパイラでコンパイルするとバグがあるたびにソース・プログラムをメモリにLOADしてバグを修正し、それをまたSAVEしてから新たにコンパイラをメモリに入れてコンパイルするというぐあいに、大変めんどろ作業になるからです。これではバグが10箇所以上もあるといいかげんいやになってしまいます。

そこで、ソース・プログラムをサブルーチン単位に分割して、そのそれぞれをあらかじめオン・メモリ版のコンパイラでデバッグしてしまおうというわけです。もちろん、このときにはサブルーチンの外にあるラベルについてのエラーは無視します。なお、オン・メモリ版でコンパイルで

きるプログラムの長さは、エンド・アドレスが\$1D00ぐらまでのものですから、コンパイルする前にモニターで\$35、\$36番地の内容がそれより小さいことを確認してください。デバッグが完了したら、そのソース・プログラムをASCII形式でSAVEしておきます。他のサブルーチンについても同様にして、最後にそれらをMERGEしてからASCII SAVEすれば完全なソース・プログラムができあがります。

ソース・プログラムができたら、ファイル版のKコンパイラとランタイム・ルーチンをLOADして、次のプログラムで最終的にコンパイルします。

```
10 CLEAR,&H1FFF
20 INPUT"FILE NAME";A$
30 OPEN"1",#1,A$
40 EXEC&H2000
50 CLOSE #1
60 CLEAR,&H4000;END
```

コンパイルが無事終了したら、EXEC&H5000で走らせることができますが、もしラベル・エラーがあったら、ラベルのつづりなどがまちがっていないか点検してください。また、数値のまちがいのバグはコンパイル・エラーになりませんから慎重に点検しておきましょう。特に配列やPOKE文のアドレスをまちがえると正常な動作をしなかったり、暴走したりします。

## 最後に

ファイル版のKコンパイラについての詳しいことは、「FM-8活用研究」を見てください。

SAVEしたオブジェクトをLOADするときは、あらかじめCLEAR,&H4000を必ず実行してください。そうしないと、Missing Operandエラーとなって正常に走りません。

コンパイルしたオブジェクトをSAVEするときは、SAVEM"ファイル名",&H4D00,&H7F8F,&H5000としてください。

### 参考文献

- 1) マイコン'82年5月号"PASOPIA CAVING"
- 2) FM-8活用研究

### 3次元迷路Kコンパイラ ソース・リスト

```
10 (* 3-D Maze Ver. 1.0 by H.I. *)
20 (* for FM-8 K-Compiler Ver. 1.2 *)
30 CONST DATA=$4010;CODE $1A,$40;CODE $10,$CE,$
3F,$FF
40 TIMES[];FOR I=1 TO TI;DMY=RND(1);NEXT
50 GF=1;SF=0;CF=1
60 HS=DATA-12;AK=DATA;XK=AK+2197;YK=XK+12;ZK=YK
+12;DK=ZK+12;FK=DK+10;QK=FK+124;RK=QK+124;EK=RK+
124;PJ=EK+6;QJ=PJ+5;RJ=QJ+5
70 FOR I=0 TO 5;HS(I)=-10000;NEXT
80 START;SC=0;MC=0;F1=1;F2=1
90 FOR I=DATA TO DATA+$AEF;POKE I,0;NEXT
100 XK(0)=-1;XK(3)=1;YK(1)=-1;YK(4)=1;ZK(2)=-1;
ZK(5)=1;DK(0)=1;DK(1)=5
110 MAZE[];INIT[];GAME[];SCORE[];GF=GF+1
120 LOC[6,9];?"PLAY AGAIN ? (Y/N)";REPEAT;I=GET;
UNTIL I<>0;IF I='N' OR I='n' THEN CODE $1C,$40;
END ELSE BEL[];GOTO START;FI
130 (* SUB *)
140 MAZE;(* Make Maze *)
150 FOR X=2 TO 10 STEP 2;FOR Y=2 TO 10 STEP 2;A
CX,Y,0,3;ACX,Y,12,3;NEXT;NEXT
160 FOR Y=2 TO 10 STEP 2;FOR Z=2 TO 10 STEP 2;A
CX,Y,Z,3;AC12,Y,Z,3;NEXT;NEXT
170 FOR Z=2 TO 10 STEP 2;FOR X=2 TO 10 STEP 2;A
CX,0,Z,3;ACX,12,Z,3;NEXT;NEXT
```

```
180 X=RND(5)*2+2;Y=RND(5)*2+2;Z=RND(5)*2+2;ACX,
Y,Z,1;PK(0)=X;QK(0)=Y;RK(0)=Z;N=0
190 FOR I=0 TO 123
200 LOOP;K=RND(N+1);X=PK(K);Y=QK(K);Z=RK(K);B=1;
T=0
210 IF A1CX-2,Y,Z)=0 THEN EK(T)=0;T=T+1;B=0;FI
220 IF A1CX,Y-2,Z)=0 THEN EK(T)=1;T=T+1;B=0;FI
230 IF A1CX,Y,Z-2)=0 THEN EK(T)=2;T=T+1;B=0;FI
240 IF A1CX+2,Y,Z)=0 THEN EK(T)=3;T=T+1;B=0;FI
250 IF A1CX,Y+2,Z)=0 THEN EK(T)=4;T=T+1;B=0;FI
260 IF A1CX,Y,Z+2)=0 THEN EK(T)=5;T=T+1;B=0;FI
270 IF B=1 THEN PK(K)=PK(N);QK(K)=QK(N);RK(K)=R
K(N);N=N+1;GOTO LOOP;FI
280 D=EK;RND(T);N=N+1;PK(N)=X+XK(D)*2;QK(N)=Y+
YK(D)*2;RK(N)=Z+ZK(D)*2;A2[D,1];ACPK(N);QK(N);RK
(N),1;
290 NEXT
300 XO=RND(5)*2+2;YO=RND(5)*2+2;ZO=RND(5)*2+2;J
=RND(3)+1;IF J=1 THEN GOTO L33 ELSE IF J=2 THEN
GOTO L34 ELSE IF J=3 THEN GOTO L35;FI;FI;FI
310 L33;XO=RND(2);DO=XO*3;XO=XO*8+2;GOTO L36
320 L34;YO=RND(2);DO=YO*3+1;YO=YO*8+2;GOTO L36
330 L35;ZO=RND(2);DO=ZO*3+2;ZO=ZO*8+2
340 L36;X=12-XO;Y=12-YO;Z=12-ZO;XO=XO+XK(DO);YO
=YO+YK(DO);ZO=ZO+ZK(DO);ACXO,YO,ZO,1;X2=XO+XK(D
O);Y2=YO+YK(DO);Z2=ZO+ZK(DO);ACX2,Y2,Z2,2;J
```

```

350 'L37: D=RND(6):IF X+XK(D)<2 OR X+XK(D)>10 OR
Y+YK(D)<2 OR Y+YK(D)>10 OR Z+ZK(D)<2 OR Z+ZK(D)
>10 THEN GOTO L37 ELSE IF A3(X,Y,Z,D)=0 THEN GOT
O L37:FI;FI
360 'L38: D1=RND(6):IF MOD(D1,3)=MOD(D,3) THEN G
OTO L38 ELSE TIME0C:FI
370 'RETURN
380 '(* F & SUB *)
390 'A: AK: %1+%2*13+%3*169=%4:RETURN
400 'A1: DMY=AK:%1+%2*13+%3*169:RETURN
410 'A2: AK: X+XK(%1)+(Y+YK(%1))*13+(Z+ZK(%1))*16
9=%2:RETURN
420 'A3: DMY=AK:%1+XK(%4)+(Y+YK(%4))*13+(Z+ZK(
%4))*169:RETURN
430 'CON: HLT:J:ADDR=FCB2:CPOKE ADDR,1,X1,X2,25
,0,25,0,X3,0,SUB:J:RETURN
440 'LINE:HLT:J:ADDR=FCB2:CPOKE ADDR,%15,%6,%5,
%1#,%2#,%3#,%4#,%7:SUB:J:RETURN
450 'CONNECT:HLT:J:ADDR=FCB2:CPOKE ADDR,%16,%9,
0,4,%1#,%2#,%3#,%4#,%5#,%6#,%7#,%8#:SUB:J:RETURN
460 'SYMBOL:HLT:J:ADDR=FCB2:CPOKE ADDR,%19,%5,0
,0,%3,%4,%1#,%2#,%6,%7,%8,%9,%10,%11,%12,%13,%14
,%15,%16,%17,%18,%19,%20,%21,%22:SUB:J:RETURN
470 'LOC: HLT:J:ADDR=FCB2:CPOKE ADDR,%3,5,%12,%1
,%2,%11,%7+%80:SUB:J:RETURN
480 'HLT: CODE %B6,%FD,%05,%2B,%FB:POKE %FD05,%B
0:CODE %B6,%FD,%05,%2A,%FB:RETURN
490 'SUB: POKE %FD05,0:RETURN
500 'GAME: (* GAME ROUTINE *)
510 'L41: F=D:B=MOD(D+3,6):D2=D1:W1=MOD(D,2)
520 'L42: W2=DK:W1)+D2:L=MOD(W2,6):IF MOD(L,3)=M
OD(D,3) THEN D2=L:GOTO L42 ELSE D2=D1:FI
530 'L43: W2=6-DK:W1)+D2:R=MOD(W2,6):IF MOD(R,3)
=MOD(D,3) THEN D2=R:GOTO L43 ELSE U=D1:G=MOD(D1+
3,6):FI
540 'CONNECT:500,199,50,0,430,0,430,199,11:CONNEC
T:0,20,480,20,480,179,0,179,11
550 'LINE:0,50,20,0,6,0:LINE:480,0,430,20,0,6
,0:LINE:0,199,50,179,0,6,0:LINE:480,199,430,17
9,0,6,0:IF A3(X,Y,Z,U)=0 THEN GOTO L48:FI
560 'IF A3(X,Y,Z,L)=0 THEN LINE:0,50,20,0,1,0:
FI
570 'IF A3(X,Y,Z,R)=0 THEN LINE:480,0,430,20,0,1
,0:FI
580 'CONNECT:50,0,50,20,430,20,430,0,6:
590 'L48: IF A3(X,Y,Z,R)=0 THEN GOTO L52:FI
600 'IF A3(X,Y,Z,U)=0 THEN LINE:480,0,430,20,0,1
,0:FI
610 'IF A3(X,Y,Z,G)=0 THEN LINE:480,199,430,179,
0,1,0:FI
620 'CONNECT:480,20,430,20,430,179,480,179,6:
630 'L52: IF A3(X,Y,Z,G)=0 THEN GOTO L56:FI
640 'IF A3(X,Y,Z,R)=0 THEN LINE:480,199,430,179,
0,1,0:FI
650 'IF A3(X,Y,Z,L)=0 THEN LINE:0,199,50,179,0,1
,0:FI
660 'CONNECT:50,199,50,179,430,179,430,199,6:
670 'L56: IF A3(X,Y,Z,L)=0 THEN GOTO L60:FI
680 'IF A3(X,Y,Z,G)=0 THEN LINE:0,199,50,179,0,1
,0:FI
690 'IF A3(X,Y,Z,U)=0 THEN LINE:0,50,20,0,1,0:
FI
700 'CONNECT:0,20,50,20,50,179,0,179,6:
710 'L60: LINE:51,21,429,178,0,1,2:Y1=X:Y1=Y:Z1
=Z:H2=50:V2=20:EF=1
720 'L61: H1=H2:V1=V2:H2=H1/3*2+80:V2=V1/3*2+33
:H3=48-H1:H4=48-H2:V3=199-V1:V4=199-V2:X1=X1+X
K(F):Y1=Y1+YK(F):Z1=Z1+ZK(F):EF=EF+1
730 'IF A1(X1,Y1,Z1)=2 THEN LINE:H1,V1,H3,V3,0,5
,2:IF EF=8 THEN SYMBOL:H1+220/EF-18,96,(130/EF-
2)/10,6/EF,0,4,%45,%58,%49,%54:FI:GOTO L86:FI
740 'IF A1(X1,Y1,Z1)=0 THEN GOTO L80 ELSE LINE:H
1,V1,H2,V2,0,6,0:LINE:H3,V1,H4,V2,0,6,0:LINE:H
1,V3,H2,V4,0,6,0:LINE:H3,V3,H4,V4,0,6,0:FI
750 'L64: IF A3(X1,Y1,Z1,U)=0 THEN GOTO L68:FI
760 'IF A3(X1,Y1,Z1,L)=0 THEN LINE:H1,V1,H2,V2,0
,1,0:FI
770 'IF A3(X1,Y1,Z1,R)=0 THEN LINE:H3,V1,H4,V2,0
,1,0:FI
780 'LINE:H1,V1,H3,V1,0,6,0:CONNECT:H2,V1,H2,V2
,H4,V2,H4,V1,6:
790 'L68: IF A3(X1,Y1,Z1,R)=0 THEN GOTO L72:FI
800 'IF A3(X1,Y1,Z1,U)=0 THEN LINE:H3,V1,H4,V2,0
,1,0:FI
810 'IF A3(X1,Y1,Z1,G)=0 THEN LINE:H3,V3,H4,V4,0
,1,0:FI
820 'LINE:H3,V1,H3,V3,0,6,0:CONNECT:H3,V2,H4,V2
,H4,V4,H3,V4,6:
830 'L72: IF A3(X1,Y1,Z1,G)=0 THEN GOTO L76:FI

```

```

840 'IF A3(X1,Y1,Z1,R)=0 THEN LINE:H3,V3,H4,V4,0
,1,0:FI
850 'IF A3(X1,Y1,Z1,L)=0 THEN LINE:H1,V3,H2,V4,0
,1,0:FI
860 'LINE:H3,V3,H1,V3,0,6,0:CONNECT:H4,V3,H4,V4
,H2,V4,H2,V3,6:
870 'L76: IF A3(X1,Y1,Z1,L)=0 THEN GOTO L61:FI
880 'IF A3(X1,Y1,Z1,G)=0 THEN LINE:H1,V3,H2,V4,0
,1,0:FI
890 'IF A3(X1,Y1,Z1,U)=0 THEN LINE:H1,V1,H2,V2,0
,1,0:FI
900 'LINE:H1,V3,H1,V1,0,6,0:CONNECT:H1,V4,H2,V4
,H2,V2,H1,V2,6:GOTO L61
910 'L80: LINE:H1,V1,H3,V3,0,1,1:X1=X1+XK(B):Y1
=Y1+YK(B):Z1=Z1+ZK(B)
920 'IF A3(X1,Y1,Z1,U)=0 THEN LINE:H1,V1,H3,V1,0
,6,0:FI
930 'IF A3(X1,Y1,Z1,R)=0 THEN LINE:H3,V1,H3,V3,0
,6,0:FI
940 'IF A3(X1,Y1,Z1,G)=0 THEN LINE:H3,V3,H1,V3,0
,6,0:FI
950 'IF A3(X1,Y1,Z1,L)=0 THEN LINE:H1,V3,H1,V1,0
,6,0:FI
960 'L86:
970 'LOC[22,23]:PRINT(" ",#2,X," ",#2,Y," ",
"#2,Z," ")
980 'LKEY: TIMES:J:F2=(MC*10+TI)/6+1:IF F2>197 T
HEN F2=198:FI:LINE:486,F1,509,F2,0,4,2:FI:F1=F2
990 'IF F2>197 THEN GOTO L60 GIVEUP:FI
1000 'LOC[68,2]:? #2,TH," ",#2,TM," ",#2,TS:KY=K
EY:J
1010 'IF KY='B' OR KY='0' THEN D=B:GOTO L41 ELSE
IF KY='L' OR KY='4' THEN D=L:GOTO L41 ELSE IF K
Y='R' OR KY='6' THEN D=R:GOTO L41:FI:FI
1020 'IF KY='0' THEN SF=-1:ZM=Z:GOTO MAP1 ELSE I
F KY='U' OR KY='8' THEN D=U:D1=B:GOTO L41 ELSE I
F KY='D' OR KY='2' THEN D=G:D1=F:GOTO L41:FI:FI
1030 'IF KY='G' THEN GOTO GIVEUP ELSE IF KY='M'
OR KY='+' THEN GOTO MAP ELSE IF KY='F' OR KY='S'
THEN GOTO L90 ELSE GOTO LKEY:FI:FI
1040 'L90: IF A3(X,Y,Z,F)=0 THEN GOTO LKEY:FI
1050 'X=X+XK(F):Y=Y+YK(F):Z=Z+ZK(F):IF A1(X,Y,Z)
<>2 THEN GOTO L41:FI
1060 'CON[4,0,0]:TIMES:J:LINE:0,0,482,199,0,5,2
:SC=(1200-TI)*15-MC*300
1070 'SYMBOL[65,40,3,2,1,15,%43,%4F,%4E,%47,%52
,%41,%54,%55,%4C,%41,%54,%49,%4F,%4E,%21:LINE:60
,68,400,190,0,0,2:
1080 'LOC[8,11]:? "TIME. ",TH," ",#2,TM," ",#2,T
S
1090 'LOC[9,13]:? "SCORE ",#6,SC
1100 'L98: RETURN
1110 '(* F & S *)
1120 'MOD: DMY=%1-%1/2*%2:RETURN
1130 'TIME: HLT:J:ADDR=FCB2:CPOKE ADDR,%3D,2,0,
0,0,0:SUB:J:RETURN
1140 'TIMES:HLT:J:POKE %FCB2,%3E:SUB:J:HLT:J:TH=
PEEK(%FCB5):TM=PEEK(%FCB6):TS=PEEK(%FCB7):TI=TM*
60+TS:POKE %FCB0,%80:SUB:J:RETURN
1150 'KEY: HLT:J:ADDR=FCB2:CPOKE ADDR,%29,0:SU
B:J:HLT:J:KY=PEEK(%FCB3):POKE %FCB0,%80:SUB:J:DM
Y=KY:RETURN
1160 'BEEP: POKE %FD03,65:RETURN
1170 'BEL: POKE %FD03,%81:FOR I=0 TO 2500:NEXT:P
OKE %FD03,%80:RETURN
1180 '(* GIVE UP *)
1190 'GIVEUP:
1200 'BEL:CON[1,40,0]:SC=-10000:SYMBOL[30,81,9
,6,7,6,%49,%20,%77,%69,%6E,%2E]
1210 'IF GF>0 THEN GF=GF-1:FI
1220 'GOTO L98
1230 'MAP: (* M A P *)
1240 'LOC[65,12]:? "FLOOR from ":ZS=INPUT:IF Z
S<0 OR ZS>12 THEN BEEP:GOTO MAP:FI
1250 'L2:LOC[65,13]:? "FLOOR to ":ZE=INPUT:IF Z
E<0 OR ZE>12 THEN BEEP:GOTO L2:FI
1260 'ZM=ZS
1270 'REPEAT
1280 'MAP1:CF=-CF:C=(CF+9)/2:J=MOD(ZM,10)+%30:IF
ZM<10 THEN I=%20 ELSE I=%31:FI
1290 'LINE[543,133,610,140,0,2,2]
1300 'SYMBOL[543,133,2,1,7,4,1,J,%20,%461
1310 'FOR YM=0 TO 12:FOR XM=0 TO 12:W=A1[XM,12-Y
M,ZM]
1320 'IF W=0 OR W=3 THEN SYMBOL[XM*8+525,YM*4+13
8,1,1,C,1,%83:GOTO NXT:FI
1330 'IF W=2 THEN SYMBOL[XM*8+525,YM*4+138,1,1,7
,1,%83:GOTO NXT:FI

```

```

1340 'IF W=1 THEN SYMBOL[XM*8+525,YM*4+138,1,1,0
,1,$B3];FI
1350 'IF XM=X AND 12-YM=Y AND ZM=Z THEN SYMBOL[X
M*8+525,YM*4+138,1,1,7,1,$A1];FI
1360 'NXT: NEXT;NEXT;BELC]
1370 'MC=MC+1;IF SF=1 THEN SF=0;GOTO LKEY;FI
1380 'ZM=ZM+SGN(ZE-ZS)
1390 'UNTIL ZM=ZE+SGN(ZE-ZS)
1400 'GOTO LKEY
1410 'INIT;
1420 'CON[1,80,1];LINE[100,3,3,0,15,0,6,2];SYMBOL
L[114,6,2,1,2,16,$33,$2D,$44,$69,$6D,$65,$6E,$73
,$69,$6F,$6E,$20,$4D,$61,$7A,$65]
1430 'J=MOD[Z2,10]+$30;IF Z2<10 THEN I=$20 ELSE
I=$31;FI
1440 'LOC[65,10];PRINT(" ",#2,X2," ",",",#2,Y2," ",
",",#2,Z2,"");LINE[170,192,310,199,0,2,2];SYMBOL[1
77,193,2,1,5,8,$45,$58,$49,$54,$20,I,J,$46]
1450 'LINE[483,0,513,199,0,2];LINE[514,74,639,
75,0,0,1]
1460 'LINE[514,92,639,94,0,0,2];LINE[514,112,639
,118,0,0,2]
1470 'LINE[514,0,639,73,0,7,1];LINE[514,76,639,9
1,0,7,1]
1480 'LINE[486,0,509,199,0,3,2];LINE[485,0,510,1

```

```

99,0,7,1]
1490 'SYMBOL[544,4,1,1,7,8,$5B,$20,$54,$49,$4D,$
45,$2E,$5D]
1500 'LOC[71,41]??"8";LOC[71,51]??"1";LOC[69,61]??"
4-5-6"
1510 'LOC[71,71]??"1";LOC[71,81]??"2";LOC[66,151]??"
[" M A P "]
1520 'LINE[514,129,639,199,0,7,1]
1530 'LINE[515,130,639,198,0,2,2]
1540 'RETURN
1550 'SCORE;(* Hi-Score *)
1560 'IF GF>5 THEN GF=5;FI
1570 'FOR I=1 TO 5;IF SC>HS(I) THEN GOTO L160;FI
;NEXT
1580 'L160: FOR J=4 TO I STEP-1;HS(J+1)=HS(J);NE
XT;HS(I)=SC
1590 'LOC[5,16];PRINT"## SCORE BEST 5 ##"
1600 'LOC[8,18];PRINT"[ 1 ] *****"
1610 'LOC[8,19];PRINT"[ 2 ] *****"
1620 'LOC[8,20];PRINT"[ 3 ] *****"
1630 'LOC[8,21];PRINT"[ 4 ] *****"
1640 'LOC[8,22];PRINT"[ 5 ] *****"
1650 'FOR I=1 TO GF;LOC[16,I+17]? #6,HS(I);NEXT
1660 'RETURN

```

## R A N D O M B O X

### マシン語ダンプ・リストの見方

最初にリストを見てください。これは\$6F00番地("\$"はその後の数値が16進数であることを表す)から\$6FFF番地までのマシン語ダンプ・リストです。左からアドレス、データ、チェック・サム、そして最下段もチェック・サムを意味しています。

アドレスはデーター1つ1つ付いているものですが、ここでは一番左にあるアドレス(最少アドレス)のみを表わしています。ここで\$6F00はデータ"34"のアドレスを示しています。データ"34"の右隣にある"37"は\$6F01番地、その隣の"E C"は\$6F02番地となっています。これらデータのリストは独立モニタ、音声モニタ、SUPER MONITOR, MEMORY CHANGERなどでとれます。

一番右端と最下段にある数値はチェック・サムと呼ばれるもので、これはプログラム・データではありません。(入力する必要はない)。このチェック・サムは、その1行のデータの合計を示しています。6F00H番地から6F0FH番地では、34H+37H+ECH+...+8DH=\$37Eとなり、下2桁を取り"E7"となっています。

このようなチェック・サムをあらかじめリストに付加する

ことによって、リストの入力ミスした箇所を素早く見つけることができます。たとえば、\$6F05番地の"83"を"82"と間違えて入力すると、横のチェック・サムは"7D"となってしまいます。また、このときのその列の最下段のチェック・サムも"13"となるので一目瞭然です。

このように長いリストのデータの1つ1つを見直さなくてもチェック・サムだけを見れば間違えて入力した箇所を発見できるようになっています。

#### リ ス ト

アドレス	データ																このデータのアドレスは6F36です。チェックサム
Addr	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6F00	34	37	EC	02	10	83	00	04	24	13	E7	BD	00	26	30	BD	:7E
6F10	00	13	AF	BD	00	0B	30	BD	00	05	BD	F2	D8	35	B7	10	:9F
6F20	00	00	00	00	77	00	00	3F	59	41	4D	41	55	43	48	49	:07
6F30	93	D3	90	90	00	34	77	7D	D4	09	86	50	BE	00	00	10	:FF
6F40	BE	3E	80	E6	8C	EE	C4	02	27	07	BE	3D	E0	10	BE	3F	:2B
6F50	60	E6	8C	EE	C4	01	27	19	EE	B9	00	00	EF	A9	80	00	:C6
6F60	EE	B9	40	00	EF	A9	40	00	EE	B1	EF	A1	4A	26	E9	20	:07
6F70	10	CE	00	00	EF	A9	80	00	EF	A9	40	00	EF	A1	4A	26	:CE
6F80	F3	FC	D0	1D	C3	00	A0	66	8C	AA	66	BC	A7	24	03	83	:1E
6F90	01	40	B4	3F	FD	D0	1D	FD	D4	0E	35	F7	00	00	00	00	:F9
6FA0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
6FB0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
6FC0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	:00
6FD0	00	00	00	00	00	00	00	F8	E7	F8	E7	F6	50	05	A0	F2	:90
6FE0	05	A0	11	0A	00	00	00	8C	4B	04	3C	DC	74	DD	11	:B7	
6FF0	D3	9C	0A	40	04	41	9C	4B	04	3C	DA	E2	70	57	70	D0	:C5
Sum:	7F	00	E6	8B	79	14		9B	9A	2B	3F	23	DF	BB	AD	B2	:09
	この中へ入力する																

この中だけ入力する

# F/M/ラ/リ-

■高畑英樹

FM-8用高速ゲームをマシン語で作ってみました。このゲームはサブCPUを倍速にしている方には非常に難しいことでしょう。スリルあふれるこのゲームを一度試してみてください。

## プログラムの説明

初期設定としてサブCPUに表示プログラムを送り、映像処理をサブCPUで、音声処理をメインCPUで出力するようにしています。

音出力はサブよりメインに高速割り込みで処理し、できるだけサブ側の負担を軽くしています。このゲームはスピード感が命ですから、この点を心掛けてプログラムしました。

その一つは表示パターンの配色は一色か二色にとどめ、表示速度を上げています。また、車の走行は道路の両側の木立ちを画面の上から下まで車のスピードに応じて常時移動させ、相対的に走行を表現しています。

エンジン音などはノイズ+パルス波形をメインで車の速度に応じた周波数を常時出力しています。BASICは画面の初期設定で、後は\$3000を入口とするマシン語に突入します。マシン語エリアは\$3000から\$3EEDまでです。

## ゲームの説明

キー操作の説明はTV画面でしています。内容は一定時間で走った距離を記録し、これを更新することです。が、あまりスピードを上げると前を走っている車に追突しやすくなります。また、ある程度スピードが出ないと方向キーは効きません。スピードは控え目に!!

他の車に当たったときとガード・レールに当たったときは、スタート地点に返されて記録はゼロになります。時間内であれば再び挑戦できます。

## 音出力について

音出力変更リストの通り外部アンプおよびスピーカーを利用すると一層迫力のあるゲームが楽しめます。パルス波に少し高周波ノイズを加えると本当にメカニカルな回転音

になるものですね。外部スピーカーだとスリップのタイヤの音までそれらしく聞こえてくるものです。カセット・インターフェイスのMIC(赤)に出力します。

## 最後に注意点を

メイン側の高速割り込みのベクタ・アドレス\$FFF6-7をゲーム・スタートと同時に書き換えています。ゲーム中[STOP]キーが押されると本来のアドレス(\$1E0)に復帰されますが、プログラム・エラーなどで復帰されずにホット・スタートさせた場合、CPUは[STOP]キーで暴走します。モニタでベクタ・アドレスを書き換えるか、電源を切る以外に方法はないので注意をしてください。周辺機器が誤動作します!

## 最後に

タイトルにふさわしく、メインもサブも一体となり瞬時も休まず、それぞれの分野でフル稼働をします。別の名を“メインもサブもフル回転”といいます。

それではFM-8“メインとサブのスピードの共演”を楽しんでください。主役はレーサーのあなたです。

SOUND出力変更リスト

変更アドレス	内部	外部
3313	03	00
331A	03	00
3327	03	00
3332	03	00
3350	03	00
3355	03	00
336E	03	00
3378	03	00
3484	03	00
348F	03	00

## BASIC プログラム・リスト

```

10 COLOR4,0:WIDTH80,20:SYMBOL(160,10),"*** GAME Message ***",2,2,4
20 LOCATES,6:PRINT"コノ ゲーム ハ キメラレタ イッテイ ノ シンカン デハ ハシヤツ キョリ ノ キロク ラ コウシン スルコトカ" モ
   クテキ デス。"
30 LOCATES,8:PRINT"レーシング"カー ノ ソウサ ハ カースル イト"ウキー ノ ソレゾレ ノ ホウコウ カ" TV カ"メン ノ フ"レイ
   ヤーカー ノ ホウコウ ニ タイオウ シテイマ。ソレゾレ ノ ホウコウ ノ テイシ ハ ソノタ ノ キー(Space)デ テイシ シマス。"
40 COLOR2,0:LOCATES,12:PRINT"プログラム ラ SUB CPU ニ オクワッテ シ"ッコウ シテイル タメ STOPキー カ" ハタ
   ラキマゼン サイキト"ー ハ STOPキー"ヲ オシナカ"ラ RESETキー ラ オシテ STOPキー ラ ハナシテ クダ"サイ。 STOPキー ダク"ラ
   オスト オトカ" デ"ナフ ナリマス。":COLOR4,0
50 LOCATE50,18:PRINT"HIT eny key";:I$=INPUT$(1)
60 CLS:LINE(350,20)-(560,189),PSET,1,BF
70 SYMBOL(350,10),"FM",3,1,3:SYMBOL(420,10),"Rally",2,1,6:SYMBOL(380,28),"Indica
   tor",2,1,7
80 X=364:Y=40:GOSUB90:X=452:GOSUB90:X=516:GOSUB90:GOTO100
90 LINE(X,Y)-(X+31,Y+120),PSET,0,BF:LINE(X+3,Y+1)-(X+28,Y+119),PSET,5,BF:LINE(X+
   10,Y+5)-(X+21,Y+115),PSET,0,BF:LINE(X,Y)-(X+10,Y+5),PSET,0:LINE(X+31,Y)-(X+21,Y+
   5),PSET,0:LINE(X+10,Y+115)-(X,Y+120),PSET,0:LINE(X+21,Y+115)-(X+31,Y+120),PSET,0
:RETURN
100 SYMBOL(355,165),"RECORD IN RECORD TIME",1,1,7
110 I$="10":FORX=45TO150STEP11:SYMBOL(415,X),I$,1,1,7:I=VAL(I$):I=I-1:I$=STR$(I)
:NEXT:SYMBOL(415,X-2),"Km",1,1,7
120 EXEC#H3000:END

```

## マシン語ダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3000	B6	FD	05	28	FB	1A	50	B6	80	97	FD	05	CE	FC	80	30	:B1
3010	BD	00	96	C6	4A	A6	80	A7	C0	5A	B7	F9	7F	FD	05	CC	:B6
3020	00	00	83	00	01	26	F8	33	8D	00	C7	BE	D0	F0	30	88	:33
3030	CE	B6	00	C6	32	8D	31	7F	FD	05	7F	FD	05	B6	FD	05	:D1
3040	2B	F8	33	8D	05	9E	BE	C0	00	30	88	CC	B6	4F	C6	34	:2A
3050	8D	16	86	01	B7	FC	80	7F	FD	05	7F	FD	05	CC	00	00	:12B
3060	B3	00	01	26	F8	16	01	FD	DD	00	B6	FD	05	2B	F8	D7	:4A
3070	01	86	00	B7	FD	05	B7	FD	05	3A	BF	FC	FE	10	BE	FC	:06
3080	CA	A6	C0	A7	A0	0A	01	26	F8	F7	FC	80	7F	FD	05	7F	:13
3090	FD	05	B6	FD	05	2B	F8	B6	80	B7	FD	05	B7	FD	05	B6	:0E
30A0	FC	B0	26	E8	0A	D6	C7	39	00	00	3F	59	41	4D	41	:21	
30B0	55	43	48	49	93	D3	BF	90	B7	D4	0A	B6	D4	0B	B6	D4	:5F
30C0	09	B6	D4	0A	F7	D3	B0	7F	D3	B0	7F	D3	B0	B6	D4	0A	:A7
30D0	B6	D3	B0	27	F8	B1	01	27	15	B7	D4	0A	B7	D4	0A	FE	:0E
30E0	D3	FE	BE	D3	CA	E6	B0	E7	C0	4A	26	F9	20	D6	7E	C0	:A6
30F0	00	12	E7	B9	B0	00	E7	B9	B0	00	E7	B9	B0	00	39	:2D	
Sum:	F7	21	3D	E2	38	EA	74	8F	42	08	61	B2	F3	18	6A	DB	:D9

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3100	00	00	86	12	97	F0	B6	80	97	F3	97	F7	97	F8	39	B6	:18E
3110	52	31	BD	00	3A	F7	23	20	0A	B6	A2	31	BD	00	30	A7	:F8
3120	23	20	0B	AE	C1	A6	C0	BD	12	6D	C4	26	F8	39	34	50	:F2
3130	8D	F3	35	10	30	88	50	35	40	20	EA	C6	08	30	C3	D8	:F2
3140	00	1F	02	C6	0B	34	10	A6	A0	34	0A	BD	10	8D	0E	30	:119
3150	B9	00	A2	35	04	5A	26	EF	35	10	A6	B1	39	46	56	57	:168
3160	46	56	57	46	56	57	46	56	57	9D	F0	30	1F	39	AE	C1	:5D
3170	A6	C0	BD	07	6D	C4	26	F8	A6	C0	39	C6	08	30	C3	D8	:18E
3180	00	1F	02	34	10	B6	08	E6	A0	9D	F0	30	88	50	4A	26	:7F
3190	F6	35	10	A6	80	39	33	BD	00	C1	BE	1F	52	17	FF	62	:92
31A0	B6	53	97	F0	0F	F3	17	FF	70	33	BD	00	AE	8E	1F	52	:55
31B0	17	FF	4F	04	F3	04	F7	04	F6	16	FF	5D	33	8D	00	88	:13
31C0	17	FF	3F	0F	F3	04	F7	16	FF	A3	33	BD	00	6E	17	FF	:4F
31D0	31	B6	53	97	F0	0F	F3	0F	F7	0F	FF	17	FF	31	33	BD	:AA
31E0	00	5A	17	FF	1D	04	F7	04	F3	16	FF	23	33	8D	00	40	:B7
31F0	17	FF	0F	B6	53	97	F0	0F	F3	0F	FF	0F	FF	17	FF	0F	:18C
Sum: 69 FD BB 11 76 D2 75 F3 AC 26 EB 9A 7C 7F E6 B5 :9C																	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3200	33	8D	00	2C	17	FE	FB	04	F3	04	F7	04	FB	16	FE	FB	:00
3210	9E	37	CD	08	86	08	97	00	C6	FF	B6	18	6F	89	B0	6F	:01
3220	89	40	00	E7	80	4A	26	F3	30	88	38	0A	00	26	EA	39	:DB
3230	37	D0	53	74	61	72	74	20	20	4F	48	00	37	CE	54	72	:8A
3240	79	20	61	67	61	69	6E	20	21	21	00	37	D2	20	48	69	:D5
3250	74	20	61	6E	79	20	6B	65	79	20	00	47	41	40	45	20	:9F
3260	4F	56	45	52	30	8D	00	D9	9F	08	30	8D	00	08	B6	:00	
3270	FF	F6	1C	00	16	00	8D	B6	F4	04	84	02	26	0D	8E	01	:B3
3280	0E	BF	FF	F6	30	8D	00	B9	16	00	76	B6	80	B7	FD	05	:55
3290	B7	FD	05	B6	FC	FF	7F	FD	05	7F	FD	05	81	00	26	01	:14
32A0	38	B1	01	26	09	7F	FD	03	8D	00	58	20	53	B1	02	:76	
32B0	26	10	0F	04	CC	03	07	DD	05	CC	03	00	DD	07	DD	09	:9A
32C0	20	3D	B1	03	26	09	CC	01	FF	DD	07	DD	09	20	30	B1	:77
32D0	0E	24	0F	97	00	86	0D	90	00	C6	13	38	B6	01	DD	09	:7E
32E0	20	1D	B1	0E	26	06	30	8D	00	1C	20	15	B1	0F	27	01	:BE
32F0	3B	30	8D	00	02	20	0A	BE	33	9C	BD	33	BE	20	05	9E	:F2
Sum:	3D	58	F5	B2	3F	CD	1E	5A	34	78	8E	72	4D	65	19	9C	:D6

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3300	08	AF	61	38	20	FE	BE	30	00	B6	96	97	00	A6	B0	2B	:36
3310	05	7F	FD	03	20	05	C6	41	F7	FD	03	C6	06	5A	26	FD	:F0
3320	48	26	EC	B6	41	B7	FD	03	10	8E	00	20	31	3F	26	FC	:28
3330	7F	FD	03	10	8E	00	20	31	3F	26	FC	0A	00	26	CE	20	:1C
3340	01	39	BE	30	00	10	9E	0A	06	8A	20	A7	0F	C6	41	F7	:FD
3350	03	20	03	7F	FD	03	D6	06	5A	26	FD	48	26	EC	BC	37	:1B

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3360	00	26	03	8E	30	00	31	3F	26	DE	C6	41	F7	FD	03	10	:69
3370	9E	07	31	3F	26	FC	7F	FD	03	C6	05	10	9E	07	31	3F	:A6
3380	26	FC	5A	26	F6	DC	07	10	93	09	27	B9	24	07	C3	00	:F5
3390	01	DD	07	20	B0	B3	00	01	DD	07	20	A9	00	64	00	64	:AA
33A0	00	02	00	01	00	01	00	01	00	00	96	00	05	00	18	00	:BB
33B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	AE	02	:6C
33C0	02	33	8D	02	01	20	08	01	2C	00	96	00	00	00	00	EC	:9C
33D0	0E	ED	2A	EC	88	10	ED	AB	17	16	00	18	BD	33	8D	46	:19
33E0	BD	3C	26	F8	39	BD	2A	BD	79	8D	33	26	F8	39	BD	21	:A2
33F0	BD	52	BD	2A	26	F8	39	BD	18	BD	2B	BD	1C	BD	45	BD	:52
Sum: CA 60 DD A7 F0 DE F4 C0 B3 57 C2 5C 2B 17 A4 97 :D5																	

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
3400	1D	26	F4	39	BD	0B	BD	3C	BD	0F	BD	1A	BD	10	26	F4	:CB
3410	39	EC	84	ED	24	EC	02	ED	26	EC	0C	ED	28	39	EC	2A	:17
3420	B3	00	01	ED	2A	39	BD	58	EC	24	A3	06	24	01	39	ED	:BD
3430	24	EC	26	A3	0A	24	01	39	ED	26	EC	28	B3	00	01	ED	:D9
3440	28	26	E3	39	BD	3A	EC	24	E3	04	28	01	39	ED	24	EC	:87
3450	26	E3	08	ED	26	28	01	39	ED	28	B3	00	01	ED	28	26	:59
3460	E3	39	BD	1C	EC	24	A3	06	24	01	39	ED	24	EC	26	E3	:E2
3470	08	ED	26	28	01	39	ED	28	B3	00	01	ED	28	26	E3	39	:6C
3480	B6	41	B7	FD	03	EC	24	B3	00	01	24	F8	4F	B7	FD	03	:37
3490	EC	26	B3	00	01	24	F8	39	00	00	00	00	00	BD	36	17	:00
34A0	E2	BD	3C	26	F7	39	BD	2C	BD	3F	BD	35	26	F8	39	BD	:2E
34B0	23	17	00	93	BD	2B	26	F7	39	BD	19	BD	2C	BD	1D	17	:F8
34C0	00	85	BD	1D	26	F3	39	BD	08	BD	7C	BD	0F	BD	1A	BD	:F2
34D0	10	26	F4	39	EC	84	ED	24	EC	02	ED	26	EC	0C	ED	28	:F2
34E0	39	EC	2A	B3	00	01	ED	2A	39	BD	06	3A	06	EC	0A	36	:7B
34F0	06	BD	3C	35	06	B3	00	01	26	0E	EC	24	B3	00	01	26	:7C
Sum:	FC	5C	9C	F4	25	B2	7E	00	1E	C8	32	D8	84	2D	1D	D5	:00

```

36C0 03 97 AA 96 04 26 21 96 E0 27 07 BE 00 FA 30 1F :A0
36D0 26 FC 96 E5 27 05 CC FF B0 DD 40 0C EE 96 EE 84 :63
36E0 01 10 27 03 C1 16 01 DE 0C EF BD D2 0E 16 FF 48 :EA
36F0 96 04 26 2B 0A D8 26 1D B6 07 97 D8 0C D7 96 D7 :5C

```

Sum: ED E8 77 45 3D 58 D3 28 1A D7 AD 5E 76 55 4D 84 :B9

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3700 84 01 27 0E 96 DD 26 05 BD D1 EA 20 58 BD D1 C8 :9E
3710 20 53 BD D2 0E CC 08 88 B3 00 01 26 FB 20 46 86 :30
3720 50 97 CB 0F ED 96 15 84 3F 8A 40 97 EA 96 0D 84 :BE
3730 3F 8A 40 97 EB 86 30 97 E9 BD D2 0E 96 ED 10 26 :17
3740 FF 79 0A 2D D6 2D C4 01 10 26 01 EB 96 D5 90 CA :5B
3750 97 D5 24 11 86 B4 97 D5 9E D0 30 88 B0 63 84 63 :67
3760 89 40 00 9F D0 0A D4 26 14 86 0F 97 D4 9E D2 30 :F0
3770 88 50 6F B4 8C 70 B2 10 27 FF 25 97 D2 96 04 B1 :60
3780 1C 26 1F 96 CA B0 03 10 2B 00 D9 17 01 14 96 E2 :FC
3790 80 20 10 27 00 CE DC 21 C3 00 01 DD 21 0C E2 16 :68
37A0 00 C2 81 1D 26 2C 96 CA B0 03 10 2B 00 B6 17 00 :9D
37B0 F1 96 E2 B0 10 10 27 00 AB DC 21 B3 00 01 DD 21 5A :50
37C0 0A E2 16 00 9F 87 D3 FF B6 D4 04 B6 D4 04 B6 D4 :D0
37D0 04 39 81 1E 26 48 0E 1C C1 48 10 27 00 B6 0A 2E :3A
37E0 96 2E 84 01 27 2C CB 06 D7 1C 0C CA 96 CA 8D 6F :E6
37F0 D4 96 E4 27 07 DC 3E C3 00 50 DD 3E 96 E3 27 07 :6B

```

Sum: DF D0 1D B7 27 B1 A5 C3 BB FA 6A 18 E1 DA BD 85 :C4

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3800 DC 3C C3 00 50 DD 3C 96 E5 27 07 DC 40 C3 00 50 :1C
3810 DD 40 17 00 BD DC 21 B3 00 F0 DD 21 20 46 81 1F :55
3820 26 42 D6 1C 27 3E 0A 2F 96 2F 84 01 27 2D 83 00 :19
3830 06 D7 1C 0A CA 96 CA 4C BD 88 96 E4 27 07 DC 3E :93
3840 83 00 50 DD 3E 96 E3 27 07 DC 3C 83 00 50 DD 3C :99
3850 96 E5 27 07 DC 40 B3 00 50 DD 40 BD 45 DC 21 C3 :47
3860 00 F0 DD 21 9E 21 33 BD 05 D0 86 1E 97 00 EC C4 :2D
3870 ED 89 80 00 ED 84 EC 42 ED 89 80 02 ED 02 30 88 :54
3880 50 33 44 0A 00 26 E7 9E 21 30 89 40 01 33 8D 06 :5D
3890 21 86 1E 97 00 EC C1 ED 84 30 88 50 0A 00 26 F5 :A7
38A0 20 24 9E 21 86 1E 97 00 CC FF FF EC 00 00 ED 84 :47
38B0 ED 02 EF 89 40 01 ED 89 80 00 ED 89 80 02 30 88 :AE
38C0 50 0A 00 26 E9 39 BE 83 2A DC 18 B3 00 12 25 04 :92
38D0 86 FF 97 00 DC 1D 93 18 03 1F 24 0F DD 00 DC :4F
38E0 1F 93 00 DD 0C DC 1F C3 01 68 93 00 1F DD 10 :65
38F0 86 B4 97 00 EC C4 2A 08 86 05 97 DE DE 1F EC C4 :60

```

Sum: E4 22 BD 57 EA 2F 4C 07 03 1E E1 7E 0E B1 BB C0 :3D

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3900 ED 84 ED 88 1C EC 42 ED 02 ED 88 1E EC 44 ED 04 :D3
3910 ED 88 1A 96 DE 2A 0A CC FB 1F A7 06 E7 88 19 20 :6F
3920 05 6F 06 6F 88 19 0A DE 30 88 50 33 46 0A 00 26 :23
3930 C3 DD 16 FE 08 9E 23 96 32 27 17 02 D2 DC 23 D3 :7E
3940 3C DD 23 1F 01 17 02 FC 96 E0 10 27 00 A7 CC E0 :71
3950 00 DD 23 0F 3D 0F E3 96 15 84 0F 4C 97 EA 16 00 :5F
3960 94 17 02 E0 96 E3 27 11 0A E6 10 26 00 87 86 03 :74
3970 97 E6 CC FF 10 DD 3C 20 7C 0A EA 26 78 0C EA 96 :2B
3980 E4 26 02 0C EE 96 CA B0 08 28 33 8E 79 20 D6 15 :5E
3990 C4 03 CB 13 CB 03 96 E5 27 07 D1 E5 25 03 5A 20 :74
39A0 F9 C0 03 C1 11 25 4E D7 E3 3A 9F 23 96 CA B0 05 :9C
39B0 D6 EE C4 01 27 01 4C D6 CB DD 3C 20 37 8E B8 :94
39C0 10 D6 15 C4 03 CB 13 CB 03 96 E5 27 07 D1 E5 25 :F2
39D0 03 5A 20 F9 C0 03 C1 11 25 1B D7 E3 3A 9F 23 86 :87
39E0 09 90 CA D6 EE C4 01 27 01 4A D6 CB DD 00 4F :68
39F0 5F 93 00 DD 3C 9E 25 96 3F 27 27 17 02 11 DC 25 :1C

```

Sum: FB 72 B2 F3 E2 27 28 42 C7 DA E8 D6 CE 58 9D AA :51

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3A00 D3 3E DD 25 1F 01 17 02 38 96 E0 10 27 00 97 CC :97
3A10 E0 00 DD 25 0F 3F 0F E4 96 15 84 0F 4C 97 EB 16 :A5
3A20 00 84 17 02 1F 96 E4 27 0F 0A E7 26 79 86 03 97 :1C
3A30 E7 CC FF 10 DD 3E 20 6E 0A EB 26 6A 0C EB 96 CA :47
3A40 B0 08 2B 2F 8E 79 20 D6 15 C4 03 CB 1A C0 04 D1 :35
3A50 E5 24 03 5C 20 F9 CB 04 C1 20 24 4A D7 E4 3A 9F :33
3A60 25 96 CA B0 04 D6 EE C4 01 27 01 4A D6 CB DD 00 :FB
3A70 3E 20 33 BE B8 10 D6 15 C4 03 CB 1A C0 04 D1 E5 :BF
3A80 24 03 5C 20 F9 CB 04 C1 20 24 1B D7 E4 3A 9F 25 :44
3A90 B6 08 90 CA D6 EE C4 01 27 01 4C D6 CB DD 00 :A0
3AA0 4F 5F 93 00 DD 3E 9E 27 96 41 27 57 0A AA 26 2C :7C
3AB0 86 03 97 AA 96 E5 91 E2 27 22 24 11 B8 04 91 E4 :3A
3AC0 27 1A 0C E5 DC 27 C3 00 01 DD 27 20 0F B0 04 91 :41
3AD0 E3 27 09 0A E5 DC 27 B3 00 01 DD 27 17 01 30 DC :B1
3AE0 D3 40 DD 27 1F 01 17 01 5A 96 E0 10 27 00 9B :18
3AF0 CC E9 00 DD 27 0F 41 0F E5 96 15 84 0F 4C 97 E9 :FE

```

Sum: DE D1 66 32 EB 79 FC A2 70 04 C5 E8 08 94 65 98 :03

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3B00 16 00 88 17 01 3E 96 E5 27 0F 0A E8 26 7D B6 03 :C3
3B10 97 EB CC FF 10 DD 40 20 72 0A E9 26 6E 0C E9 96 :1B
3B20 CA B0 08 2B 30 8E 78 B0 D6 15 C4 03 96 0C B4 01 :0C
3B30 27 01 5C CB 16 C0 04 D1 E3 25 50 CB 07 96 E4 27 :05
3B40 04 D1 E4 24 46 C0 03 D7 E5 3A 9F 27 DC CA B0 03 :CB
3B50 3D DD 40 20 36 8E BB 60 D6 15 C4 03 96 0C B4 01 :32
3B60 27 01 5C CB 16 C0 04 D1 E3 25 20 CB 07 96 E4 27 :95
3B70 04 D1 E4 24 16 C0 03 D7 E5 3A 9F 27 B6 0A 90 CA :5C
3B80 D6 CB 3D DD 00 4F 5F 93 00 DD 40 9E 21 30 89 7F :10
3B90 B1 A6 80 43 26 40 A6 84 43 26 38 30 89 01 3E A6 :EC
3BA0 84 43 26 32 A6 03 43 26 2D 30 89 06 E0 A6 84 43 :6A
3BB0 26 24 A6 03 43 26 1F 30 89 01 91 A6 80 43 26 16 :68
3BC0 A6 B4 43 26 11 96 E2 B1 10 27 0B B1 20 27 07 B6 :34
3BD0 03 97 EC 16 FC F0 B6 0E B7 D3 FF B6 D4 04 B6 D4 :BD
3BE0 04 0A EC 10 26 FC DF B6 0F 17 FB D9 B6 0A F7 D4 :E6
3BF0 08 BE FA 00 30 01 26 FC F6 D4 08 BE FA 00 30 01 :6E

```

Sum: F0 74 BA E0 71 72 EB B3 9A 1A CB 10 AE F0 A4 63 :B3

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3C00 26 FC 4A 26 E9 BE 00 00 30 1F 26 FC 16 FA 50 B6 :60
3C10 1E 97 00 C6 50 CE FF FF BC B3 20 24 0C FF E0 00 :D5
3C20 FF E0 00 3A 0A 00 26 F0 39 BC BB 60 25 0C FF E0 :29
3C30 00 FF E0 00 3A 0A 00 26 F0 39 EF 84 EF 02 3A B0 :1A
3C40 00 26 E6 39 33 BD 01 F2 B6 1E 97 00 97 E0 EC B3 :B9
3C50 20 24 14 EC C4 FD E0 00 EC 42 FD E0 00 30 88 50 :FB
3C60 33 44 0A 00 26 E8 39 CB BB 60 25 14 EC C4 FD E0 :35
3C70 00 EC 42 FD E0 00 30 88 50 33 44 0A 00 26 E8 39 :DB
3C80 EC C4 ED 84 EC 42 ED 02 0F E0 30 88 50 33 44 0A :B6
3C90 00 26 D4 39 B6 84 97 00 C6 FF BE 03 30 86 09 97 :B0
3CA0 01 B6 FF A7 B9 80 00 A7 B9 40 A7 80 ED A7 80 ED :C3
3CB0 00 6F B9 40 00 6F B9 40 01 ED B1 0A 01 26 E7 :A5
3CC0 B9 80 00 A7 B9 40 00 A7 B9 30 BB 3D 0A 00 26 DC :96
3CD0 39 00 01 80 00 C2 63 00 C3 00 01 31 BC 00 0F :5E
3CE0 00 0C B0 03 3F FC 02 02 C0 00 FF FF 00 01 80 :38
3CF0 0F F0 00 01 B0 00 1F B0 00 00 00 00 7F FE 00 :14

```

Sum: 54 47 6A 14 BD BB FE A5 0B 56 B3 AB CF CC DB E1 :47

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3D00 00 01 FF FF 80 00 00 0F FF FF E0 00 00 00 1F :7F
3D10 00 C0 03 00 3F FE 00 30 0C 00 7F FF 00 0B 10 :D3
3D20 FF FF C0 04 20 07 FF FF F0 C4 23 1F FF FF FC 32 :09
3D30 4C 7F C1 B3 FF 0A 00 01 B0 0F C5 A3 00 03 C0 :98
3D40 00 33 CC 00 07 E0 00 0F 00 00 3C 00 00 00 E0 :01
3D50 7E 00 00 00 00 00 3C 00 00 00 00 00 00 FF 00 :C0
3D60 E0 01 FF B0 00 1F FB 03 FF C0 00 7F FE 01 FF :B0
3D70 00 7F FE 07 FF E0 00 FF FF 1F FF B0 7F FE 7F :73
3D80 18 FE 00 1F FB 00 18 00 00 01 B0 00 1B 03 C0 :A4
3D90 C0 00 3C 07 E0 07 E0 00 FF 0F 0F 0F 06 00 07 :D4
3DA0 E0 00 00 0F 00 0F 0F 0F 0F 0F 0F 1F B8 20 CF :17
3DB0 B0 3F FC 11 FE 3F C0 0F 0F 0F 3F C0 1F 1F 05 :EA
3DC0 FE 3F C0 3F C0 65 FE 3F C0 7F FE 15 FE 7F E0 :08
3DD0 FE 00 FE 7F E0 7F FE 1D FE 7F 3F FE 1D FE 7F :34
3DE0 E0 01 B0 00 FC 3F C0 03 C0 00 7B 1F B0 0F E0 :15
3DF0 30 0F 00 00 00 18 7B 06 00 7F 00 04 FC 06 01 :5A

```

Sum: ED BB C2 11 92 7E 5F F3 CF D7 10 3E D5 B0 BE 00 :84

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
3E00 C0 C4 00 06 03 FF E0 32 30 06 01 FF C0 0A 47 :F4
3E10 07 FF FB 09 4B 3F C3 FF F0 05 90 00 0F FF FC :E6
3E20 20 00 3F FF FF 03 C0 00 00 0C 00 03 E0 00 1E :2D
3E30 00 00 00 00 00 3F 00 00 00 80 FF FC 3F FF FB :EF
3E40 1F FF FF FB 1F FF FF FB 1F FF FC 7B 1E 3F FB :41
3E50 0C 1F FB 00 00 1F FB 30 0C 1F FC 70 0E 3F FF :3D
3E60 0F FF FF E0 07 FF FF E1 B7 FF FF E2 47 FF :61
3E70 47 FF FF E2 47 FF FF C4 23 FF FF C4 23 FF :FA
3E80 23 FF FF CC 33 FF B1 CC 33 B1 00 CC 33 00 00 :2B
3E90 30 00 00 04 20 00 00 C3 C3 00 B1 C0 03 B1 FF :5E
3EA0 03 FF FF C0 03 FF FF C0 03 FF FF E0 0F FF :58
3EB0 0F FF 03 C0 07 E0 07 E0 07 E0 07 E0 0F 0F :68
3EC0 0F 0F 0F 0F 0F 0F 1F FB 1E 7B 1D B8 1D B8 :29
3ED0 3B DC 3B DC 3B DC 3C 3C 3C 3C 3C 3C 3C 3C :58
3EE0 3C 3C 3F FC 3F FC 3F FC 3F FC 1F FB 0F 0F :79

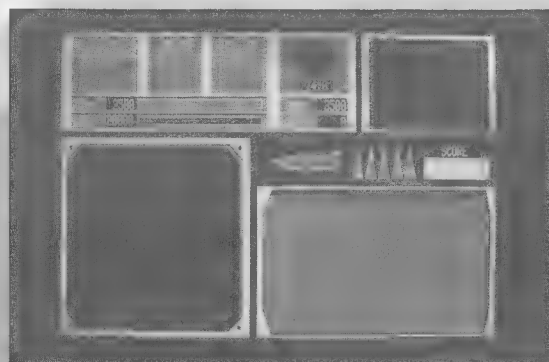
```

Sum: 53 E4 B6 E0 9D 42 70 ED B5 53 7C 54 2F 68 9A 33 :15

# 日本復活



■マイクロ・テクノロジー研究所



## I

## 日本復活の背景

2001年のある日、東京江戸川区に金属を食料として増殖する恐ろしい宇宙アメーバが小隕石に附着して飛来しました。彼らはたちまちにして、車、ビルなどを食料とし日本全国にアメーバ拠点を拡大していったのです。宇宙アメーバの武器は、人間に激しい下痢を起こさせるバイ菌と、金属を腐蝕させる恐ろしい毒ガスです。あなたはマッハ2.5で飛行するアメーバ攻撃メガの司令官として鳥島に設置されたコントロール・タワーから上手に攻撃メガロを操作してください。

## II

## 武器と種類

### A

### DDT (都市散布用)

これは1950年代に使われた殺虫剤で、宇宙アメーバ駆除に絶大な効果を示します。散布範囲は、アメーバに占拠された攻撃都市に限定され、アメーバ基地に対しては使えません。その射程は、目標に対し100km以内です。また、一度

散布した都市にDDTを発射すると人間に悪影響が出るため減点されます。

### B

### キンチョールン (基地噴射用)

キンチョールンは、ヘビーメタル研究所の大屋女史によって開発されたウルトラ殺虫剤で、基地に潜む宇宙アメーバを退治することができます。その射程は目標に対し200km以内です。

## III

## コマンドの説明

- PLー飛行 (60分までの分単位で入力)
- BRーバリア (敵攻撃に対し防衛する)
- TNーターン (敵攻撃を回避する)
- ATーアタック (キンチョールン噴射命令)
- GOーゴー (DDT散布命令)
- TGーターゲット (目的都市設定)\*
- ACーオールシティ (都市、基地までの方向、距離)
- SDーセルフデータ (残り弾数などの表示)
- CMーコマンド (コマンド一覧)
- CDーコード (DDTを散布するための命令)
- COーコース
- HIー高度 (高度入力)



H I—高 度 (高度入力)

R D—レ ー タ (短距離レーダー)

\* 宇宙アメーバ都市と基地の番コード

宇宙アメーバ都市(全10都市, DDT目標)

WN—稚内 AM—青森 TK—東京 OS—大阪 FO—福岡  
 SP—札幌 SD—仙台 KZ—金沢 ME—松江 KG—鹿児島

宇宙アメーバ基地(全20基地, キンチョルン目標)

AB—網走 ZO—蔵王 HJ—八丈 YG—山口  
 FM—根室 FS—福島 MM—松本 KC—高知  
 EM—襟裳 CB—千葉 FI—福井 NS—長崎  
 TW—十和田 SO—静岡 KM—熊本 AS—阿蘇  
 SA—佐渡 NG—名古屋 OY—岡山 TG—種子島

## IV 高得点のテクニック

このシミュレーションでは高度が約500m以下、スピードが200km/時、燃料が100ユニット以下、得点が減点によりマイナスになったときのいずれか一つに該当するとGAME

ENDとなります。また、敵攻撃によって被弾した場合、そのダメージの具合によって、一度で撃墜されることがあります。

DDTは全部で10発、キンチョルンは最初5発積まれています。このキンチョルンに対しては、敵都市攻撃の成功によって補給されます。高得点のテクニックは次の通りです。

- ① 敵基地の複数個所を横断しない。
- ② 敵都市には海洋上から一直線に進出し、攻撃後、すみやかに洋上に退避する。
- ③ バリアの能力は段階的に減少するので、むやみにバリアを使わない。
- ④ 敵攻撃によりスピードが低下した際、都市攻撃を急ぐ。この際、攻撃成功によって、スピードは元のマッハ 2.5に回復します。また、得点がある程度あり、かつ敵攻撃によってかなりスピードダウンしている際、すでに攻撃成功済みの都市が近くにあれば、その都市を再度攻撃します。これを実行するとスピードとキンチョルンは回復しますが、減点されるため墜落の危険性があります。また、最初に攻撃指定都市をねらうと高得点を稼がやすいです。

### 日本復活BASICプログラム・リスト

```

130 DIM GRA1(18),GRA2(18)
300 GOSUB6000:DIM ZK(550),C(10,2),N$(30),T(30,3)
310 RESTORE40060:FOR I=5H7E00 TO 5H7E33:READ A$:POKE I,VAL("$H"+A$):NEXT
350 FORI=1TO2:STT=0:TEM=3:ARE=15:GOSUB40010:NEXTI
351 RESTORE7500:FOR I=0 TO 30:READ N$(I),T(1,1),T(1,2):T(I,1)=T(I,1)*25:T(I,2)=T
(I,2)*25:NEXT:GOSUB 5940
353 GOSUB 30000:WIDTH40,25:CONSOLE0,25,0,0:COLOR7,0:CLS
365 SYMBOL(120,0),"アメーバ" コウキ モトヨウ ",3,3,2
370 PUT$(172,84)-(308,144),ZK,PSET,4
380 GOSUB9000:PUT$(372,84)-(508,144),ZK,PSET,5:V=0
385 LINE(120,0)-(560,30),PSET,0,BF:SYMBOL(150,0),"チ アメーバ" 33",3,3,6:GOSUB9040
998 SYMBOL(120,170),"タカ イマ コウキ オフ ラン サツセイチュウ",1,2,7:CLEAR
1000 PC1=0:NO=10:N1=20:N2=N0+N1:DIM ZK(550),CTN$(30),C(10,2),N$(30),TB(31),T(30,
3),M$(1):G$="":TKTEN$="0":TEN=0:OTEN=0
1010 ON ERROR GOTO50000:DIS=RND(55)*100
1020 ODR$="PLBRTNATTGHCOSDRDCMCDGOAC"
1025 RESTORE 1030:FOR I=0TO30:READ CTN$(I):NEXT
1030 DATA TS,WN,SP,AM,SD,TK,KZ,OS,ME,FO,KG,AB,NM,EM,TW,SA,ZO,FS,CB,SO,NG,HJ,MM,F
I,KM,OY,YB,KC,AS,NS,TG
1040 RESTORE7500:FOR I=0 TO 30:READ N$(I),T(1,1),T(1,2):T(I,1)=T(I,1)*25:T(I,2)=T
(I,2)*25:NEXT:GOSUB 5940
1070 EBD=0:EBD=1:PYO3=0:PI=3.14159:PYO2=0:X=103*25+INT(10*NRND(1)):Y=155*25+INT(
10*NRND(1)):GOSUB5910:A9=12000:A0=5000:M$(0)="ハ イキシ":M$(1)="ト クガス":C0=10:PX01=0:
S=2500
1075 PX02=0:P=5:PX03=0:F9=0:PYO1=0:T=0:XOTD1=0:XOTD2=0:XOTD3=0:YOTD1=0:YOTD2=0:Y
OTD3=0:GATY=0:OTHX=0
1080 VFLG=0:GOSUB 30000
1090 FOR I=1TO30:T(1,3)=0:NEXT:GOSUB 5940:A=INT(10000*NRND(1)):A1=A:R9=0
1100 GOSUB 5940:C=INT(RND(1)*360):GOSUB 5910:C1=C:T9=INT(RND(1)*N0+1):NN=T9:CTN=
T9:TB=0:F=200001:ENP=200001
1110 FOR I=1TO10:C(1,1)=0:NEXT:FFFZ=RND(1)*1000:FOR I=1TO5
1120 GOSUB 5910:FFFZ=FFFZ+(INT(RND(1)*1000+1)):NEXT:L9=.3
1130 N$="サツセイ モトヨウウ" "+N$(T9):SYMBOL(140,25),N$,2,2,7:CD$="DDT OPEN キーコト" "+ST
R$(FFFZ):SYMBOL(140,45),CD$,2,1,7
1230 'SYMBOL(160,190)," (Hit Any Key)",1,1,7
1240 WAIT$=INKEY$:IF WAIT$="" THEN1240
1242 CLS:SYMBOL(150,10),"GAME LEVEL",2,2,5:SYMBOL(50,50),"1...マ ツ",3,3,7:SYMBOL(
50,100),"2...タ ク",3,3,7:SYMBOL(50,150),"3...ウ メ",3,3,7:LOCATE4,23:INPUT"LEVEL":C
ST$
1244 CST=VAL(CST$):IF CST<1 OR CST>3 THEN 1242
1246 IF CST=1 THEN CST=500+INT(RND(5)*10):E=6:GOTO 1250
1247 IF CST=2 THEN CST=400+INT(RND(3)*10):E=5:GOTO 1250
1248 IF CST=3 THEN CST=300+INT(RND(2)*10):E=5:GOTO 1250
1249 CST=300:E=4
1250 STAKA=0:CFG=0:NAV$=N$(NN):OKM$="":OTARG$="":GOSUB 8000:GOSUB10510:GOSUB 850
0
2000 GOSUB2640
2010 LINE(305,120)-(624,191),PSET,1,BF
2020 AUF6=0:COLOR7:LOCATE20,14:PRINT " ":LOCATE20,14:INPUT"コマンド"
":A$:A$=RIGHT$( " "+A$,2)
2030 ODR=(INSTR(ODR$,A$)+1)/2
2040 IF ODR<>0 AND ODR=INT(ODR) THEN GO TO 2090
2050 FOR IJ=0 TO 3:BEEP:LOCATE20,14:COLOR10:PRINT "メイレイン アリマセン":FOR IJ=0 TO 200:
NEXT:LOCATE20,14:PRINT " ":NEXT:GO TO 2020
2060 FOR IJ=0 TO 3:BEEP:LOCATE20,14:COLOR10:PRINT "トリコトニ アリマセン":FOR IJ=0 TO 20
0:NEXT:LOCATE20,14:PRINT " ":NEXT:COLOR1:LOCATE20,14:PRINT " ":RETURN
2070 GOSUB5910:T0=T+3+INT(5*NRND(1)):GOTO4000
2090 ON ODR GOTO 2240,3100,2720,2860,2470,2110,2150,2690,2600,7112,2170,2300,582
0
2110 GOSUB6530
2120 IF A1<80 THEN A1=0
2130 GOSUB 5910:T0=T+13+INT(15*NRND(1)):GOTO 4000
2150 GOSUB 6560:GOTO 2130
2170 IF FL<1500 THEN LOCATE20,16:PRINT"CD キンキ チタイ":FOR BX=0TO300:NEXT BX:GOTO 20

```



## BASICプログラム・リスト

```

10
2178 IF F9=2 THEN 2220
2180 GOSUB 6580
2190 IF F9=0 THEN 2210
2200 GOSUB 6600:GOTO2130
2210 GOSUB 6610:GOTO2130
2220 GOSUB 6620:GOTO2130
2240 AUF6=1:I=0:J=0:K=0
2250 COLOR7:LOCATE20,14:INPUT"ヒコウシ カシ":A$:K=VAL(A$):IF K>60 THEN K=60
2255 IF K<1 THEN K=1
2260 K=K/60
2270 T2=T+3600*K
2280 GOTO 4000
2300 IF F9=2THEN2220
2310 IF F9=0 THEN 2430
2320 GOSUB 6640
2330 IF LEN(A$)<2THEN 2440
2340 A$=LEFT$(A$,2):GOSUB 5800:IF ((NN=-1)OR (NN>NO))THEN GOSUB2060:GOTO 2300
2350 GOSUB 5700:IF RA>100 THEN 2450
2360 GOSUB 6650
2370 IF A$<>"Y" THEN 2130
2375 FOR II=1TO10:IF TB(II)=NN THEN GOSUB 6525:GOTO 2130
2377 NEXT
2380 GOSUB 6660
2400 GOSUB 5910
2414 EBO=EBO+1:EBO=10-EBO:IF EBO>9 THEN F9=2
2417 TEN=TEN+160+INT(100*RND(1)):IF T9=NN THEN TEN=TEN+500+INT(RND(1)*100)
2418 TKTEN$=RIGHT$( " "+STR$(TEN),5):BEEP:LINE(530,82)-(630,95),PSET,S,BF:SYM
BOL(545,85),TKTEN$,2,1,1
2419 TB(EBO)=NN:TB=NN:A0=A0+50:GOSUB 5910:T0=T+6+INT(5*RND(1))
2421 IF S<2500 THEN S=2500:SPX=S*180/2500:LINE(SPX+115,46)-(295,49),PSET,O,BF:L
INE(67,44)-(111,51),PSET,O,BF:OSPD$=SPD$:SPD$=RIGHT$( " "+STR$(S),5):SYMBOL(6
8,45),SPD$,1,1,7:PAINT(293,48),4,7
2422 IF P<5 THEN P=5:GOSUB 10020
2425 GOTO 4000
2430 GOSUB 6670:GOTO2130
2440 GOSUB 6680:GOTO2130
2450 GOSUB 6900:GOTO2130
2470 GOSUB 6700
2480 GOSUB 5800
2490 IF CTN=-1 OR NN=-1 THEN GOSUB 2060:GOTO 2470
2510 GOSUB 5910:T0=T+40+INT(41*RND(1)):GOTO 4000
2520 GOSUB 6750:GOTO 2510
2530 I=0:J=480:FOR NN=(NO+1)TON2:GOSUB5700:IF RA>J THEN 2580
2550 IF T(NN,3)=2 THEN2580
2560 GOSUB 5500:IF ABS(C-AN)>45 AND ABS(C-AN)<315 THEN 2580
2570 J=RA:I=NN
2580 NEXT:NN=I:IF NN>O THEN 2510
2590 GOTO 2510
2600 GOSUB2640
2620 GOSUB 5940:T0=T+9+INT(RND(1)*11):GOTO 4000
2640 J=0:FOR I=1TON0:IF C(I,1)=0 THEN 2670
2650 J=1
2660 GOSUB 6790
2670 NEXT:RETURN
2690 IF F9<>2 THEN GOSUB 6240
2700 GOSUB6250:GOSUB5910:T0=T+20+INT(RND(1)*21):GOTO4000
2720 GOSUB 5940:A1=30+INT(150*RND(1)):IF RND(1)<.5 THEN A1=-A1
2730 A=A+A1:IF A>16000 THEN A=16000
2740 IF A<90 THEN 5000
2750 GOSUB 5910:A1=A:C1=60+INT(60*RND(1))
2760 IF RND(1)>.6 THEN C1=-C1
2770 C=C+C1:C=C+360*ABS(C<0)-360*ABS(C>360)
2780 C1=C:FOR I=1TON0:IF C(I,1)=0 THEN 2840
2790 J=(B+2*ABS(C(I,1)<10))/(C(I,2)-T):IFJ>.95 THEN J=.95
2800 IF RND(1)>J THEN 2840
2810 GOSUB 6820
2820 GOSUB 6830
2830 C(I,1)=0
2840 NEXT:GOSUB 5940:T0=T+10+INT(21*RND(1)):GOTO 4000
2860 IF P<1 THEN P=0:GOTO 2970
2870 GOSUB 6850
2880 IF A$="" THEN3000
2890 IF LEN(A$)<2THEN2980
2910 A$=LEFT$(A$,2):GOSUB 5800
2920 IF NN<(11) THEN2980
2930 GOSUB 5700:IF RA>200 THEN 2990
2932 FOR II=1TO30:IF TB(II)=NN THEN GOSUB6525:GOTO 4000 ELSE 2934
2934 NEXT
2940 TB(NN)=NN:GOSUB 6860
2950 GOSUB 6870
2960 GOSUB 5910:T0=T+3+INT(RND(1)*5):GOTO 4000
2970 GOSUB 6880:GOTO 2960
2980 GOSUB 6890:GOTO2960
2990 GOSUB 6900:GOTO 2960
3000 J=180+RND(1)*10:K=0:FOR I=1TO10:IF C(I,1)=0 THEN 3030
3010 IFC(1,2)-T>J THEN 3030
3020 J=C(I,2)-T:K=I
3030 NEXT:IF K=0THEN3090
3040 IF J>10 THEN 3090
3050 GOSUB 5910:IF C(K,1)<10 AND RND(1)>.6THEN3090
3060 GOSUB 5910:IF C(K,1)>10 AND RND(1)>.85 THEN 3090
3070 GOSUB 6930
3080 C(K,1)=0:GOTO 2960
3090 GOSUB 6930:GOTO2960
3100 FOR I=1TO10:IF C(I,1)=0 THEN 3180
3110 J=E-RND(1):IF E<1 THEN GOSUB 5910:J=RND(1)
3120 IF J>.99 THEN J=.99
3130 GOSUB 5910:IF .55>J THEN 3180
3140 LOCATE20,18:BEEP:PRINT "チ キ カシ"-"":C(I,1):" ":LOCATE20,19:PRINT M$(ABS(C(I,

```

```

1) > 10)):" エントリ " :E=E*.7:IF E<1 THEN E=.5
3150 GOSUB 6940
3160 GOSUB 6950
3170 C(I,1)=0
3180 NEXT:GOSUB 5940:T0=T+S+INT(9*RND(1))
3185 GOTO 4000
4000 T3=T0:I9=1:IF T0>T THEN4080
4010 I9=0:GOSUB 5910:T3=INT(RND(1)*360)+T
4020 FOR I=1TO 10:IF C(I,1)=0 THEN 4080
4030 IF C(I,2)<T3 THEN T3=C(I,2)
4040 IF T3>C(I,2)-10 THEN 4060
4050 IF C(I,2)-10 < T3 THEN T3=C(I,2)-10
4060 NEXT:IF (T2 =T) OR (T3 T2)THEN4080
4070 T3=T2:I9=1
4080 T0=T3-T:A2=A0+T0:GOTO4000
4090 IF A2=ABS(A-A1) THEN A=A1
4100 IF A2=ABS(A-A1) THEN A=A+A2*SGN(A1-A):IF A>16000 THEN A=16000
4110 IF C2=ABS(360*(ABS(C-C1)-180)-ABS(C-C1)) THEN 4130
4120 C=C+C2*SGN(C1-C):C=C+360*ABS(C<0)-360*ABS(C>360):GOTO 4140
4130 C=C1
4140 X=X+S*T0*SIN(C*PI/180)/3600
4150 FL=FL+INT(S*(T3-T)/3600):F=F-INT(S*(T3-T)/3600)
4160 Y=Y-S*T0*COS(C*PI/180)/3600:T=T3
4170 IF F=100 THEN 5000
4180 DE=0:CR=0:FOR I=1TO10:IF (C(I,1)=0) OR (DE=1) OR (CR=1) THEN 4270
4190 IF C(I,2) < T THEN 4250
4200 GOSUB 6200
4210 GOSUB 5910:A=A-INT(2800*RND(1)):IF A<200 THEN 5000
4220 A1=A:GOSUB 5940:F=INT(F+600*RND(1)):GOSUB 5910:S=INT(S-1900*RND(1))
4230 GOSUB 5910:IF RND(1)<.2 THEN DE=1
4240 GOTO 4270
4250 IF C(I,2)>(T+10) THEN 4270
4260 I9=1
4270 NEXT
4280 IF DE=1 THEN 6220
4281 IF S<500 THEN 6210
4282 IF A<400 THEN 6220
4283 IF F<200 THEN 6230
4284 IF F<200 THEN 6230
4290 IF F<2 THEN 4380
4300 FOR NN=1TO N0:GOSUB5700:IF RA<100 THEN4330
4310 IF T(NN,2) < 0 THEN 4350
4320 GOSUB 6980
4330 IF T(NN,2)<1 THEN 4360
4340 IF RA<100 THEN 4360
4350 GOSUB 6990
4360 IF NN=CTN THEN GOSUB10510
4370 NEXT
4380 FOR NN=(N0+1) TO N2:IF T(NN,2)=2 THEN4610
4390 IF ABS(X-T(NN,1)) < ST THEN 4610
4400 IF ABS(Y-T(NN,2)) < ST THEN 4610
4410 GOSUB 5700:IF RA<ST THEN 4610
4420 IF RA<200 THEN 4460
4430 IF T(NN,2)=1 THEN 4460
4440 T(NN,2)=1
4450 GOSUB 7000
4460 IF T(NN,2)<1 THEN 4500
4470 IF RA<200 THEN 4500
4480 T(NN,2)=0
4490 GOSUB6900
4500 GOSUB 5910:IF RND(1) < L9*(T0/300)*(A<125) THEN4610
4510 J=0:FOR K=1 TO10:IF C(K,1)=0 THENJ=K
4520 NEXT:IF J=0 THEN 4610
4530 GOSUB 5910:IF RND(1) < .5 THEN 4550
4540 GOSUB 5940:C(J,1)=INT(10*RND(1)+1):M=7000:GOTO 4560
4550 GOSUB 5910:C(J,1)=20+2*INT(RND(1)*7):M=4000
4560 GOSUB5500:L=(AN-C)*PI/180
4570 L=SOR(M^2-(S* SIN(L))^2)+SGN(PI-ABS(L))*ABS(S*COS(L))
4580 GOSUB 5700:C(J,2)=T+INT((3600*RA/L)+1)
4590 GOSUB 7020
4600 I9=1
4610 NEXT:NN=0:GOSUB5500:GOSUB5700:GOSUB8500:L=ABS(C-AN)
4620 IF RA<300 AND F<900 THEN 5000
4630 IF A<200THEN 6210
4632 IF F<900THEN 6230
4635 IF S<500THEN 6210
4637 IF TEN<0 THEN 5000
4640 IF I9=1 THEN 2000
4650 IF I9<>1 THEN 4000
5000 GOTO 7040
5500 DX=X-T(NN,1):DY=Y-T(NN,2)
5510 IF DY=0 THEN 5530
5520 AN=ATN(-DX/DY)*180/PI+180*ABS(DY<0)+360*ABS(DX<0)*ABS(DY>0):RETURN
5530 AN=90+180*ABS(DX>0):RETURN
5700 RA=SOR((X-T(NN,1))^2+(Y-T(NN,2))^2):RETURN
5800 CTN=-1:FOR SOR=0TO30:IF A<LEFT$(CTN$(SOR),2) THEN CTN=SOR
5802 NEXT SOR
5805 TCT=CTN:NN=CTN:RETURN
5820 GOSUB6700:NN=-1:FOR HEN=0TO30:IF A<LEFT$(CTN$(HEN),2) THEN NN=HEN
5830 NEXT HEN:IF NN=-1 THEN GOSUB2060:GOTO5820
5835 GOSUB 10510:GOTO 2010
5910 RANDOMIZE(VAL(RIGHT$(TIME$,2))*163)
5920 DIS=INT(SIN(DIS)*100)
5930 RETURN
5940 UUU=INT(COS(DIS)*100)
5950 RANDOMIZE(UUU)
5960 RETURN
6000 WIDTH40,25:CONSOLE0,25,0,0:COLOR7,7:CLS
6001 COLOR6:A=A+1:IF A>200 THEN 6003 ELSE PRINTCHR$(254);:GOTO 6001
6003 COLOR4:B=B+1:IF B>480 THEN 6007 ELSE PRINTCHR$(254);:GOTO 6003
6007 COLOR5:C=C+1:IF C>120 THEN 6009 ELSE PRINTCHR$(254);:GOTO 6007

```

```

6009 COLOR2:D=0+1:F D=120 THEN 6011 ELSE PRINTCHR$(254);:GOTO 6009
6011 COLOR0:E=E+1:IF E>40 THEN 6015 ELSE PRINTCHR$(254);:GOTO 6011
6015 SYMBOL(150,10), " ヒロシ フグカツ ",3,4,2:SYMBOL(100,165), "C" 1982 BY (MTI) 747077
7707 " ",2,2,1
6018 RESTORE7570:FOR HL=0T015:READ GRA1(HL):NEXT HL:FORHL=0T015:READGRA2(HL):NEX
THL
6020 LINE(580,40)-(581,41),PSET,0
6030 FOR HL=0T015
6040 LINE-(GRA1(HL),GRA2(HL)),PSET,0
6050 NEXT HL
6060 CONNECT(530,50)-(520,45),0,PSET
6080 CIRCLE(305,125),30,2,,F
6090 FAINT(350,125),1,0,2:PAINT(535,45),2,0,0:PAINT(230,120),0,0
6100 SYMBOL(80,110),"N",2,3,2
6110 SYMBOL(100,105),"F",2,3,2
6120 RETURN
6200 R9=1:C(I,1)=0:I9=1:COLOR7:LOCATE20,14:PRINT" メカ ロ ヒタ ツ " :S
TT=&HC0:TEM=2:ARE=15:GOSUB40010:STT=0:TEM=2:ARE=15:GOSUB40010:GOSUB9210:RETURN
6110 COLOR7:LOCATE20,14:PRINT" メカ ロ ツイラク " :STT=&HC0:TEM=
2:ARE=35:GOSUB40010:BEEP:GOTO 5000
6220 COLOR7:LOCATE20,14:PRINT" メカ ロハ フウツ " :STT=&HC0:TEM
=1:ARE=35:GOSUB40010:STT=0:TEM=2:ARE=25:GOSUB40010:GOTO 5000
6230 LOCATE20,14:PRINT" ヒコウ フウツ " :BEEP:STT=&HC0:TEM
=2:ARE=15:GOSUB40010:STT=&HC0:TEM=1:ARE=15:GOSUB40010:BEEP:GOTO 5000
6240 COLOR7:LOCATE20,15:PRINT" イチ エヒョウ " :N$(T9):RETURN
6250 COLOR7:LOCATE20,17:PRINT"ノコリ フグカツン":P:LOCATE20,19:PRINT"/コリ DDT " :EBD:LOCA
TE20,21:PRINT"フグカツン":STAKA
6260 RETURN
6525 LOCATE20,16:PRINT N$(NN):LOCATE20,18:PRINT"コウゴロ ヒカ" I9:STT=&HC0:TEM=1:ARE=
20:GOSUB40010:BAD=INT(200*RRND(1)):TEN=TEN-BAD:LOCATE20,19:PRINT"ケコリン " :":BAD
6526 TKTEN$=RIGHT$( " " +STR$(TEN),5):BEEP:LINE(530,82)-(630,95),PSET,5,BF:SYM
BOL(545,85),TKTEN$,2,1,1:RETURN
6530 LOCATE20,14:PRINT" " :LOCATE20,14:INPUT"コリ " :A1=A1+VAL(A1$)
:IF A1<90 THEN A1=0
6540 IF A1>16000 THEN A1=16000
6550 RETURN
6560 LOCATE20,14:INPUT"コリ " :C1=C1+VAL(C1$):IF C1<1 OR C1>360 THEN C1=1
6570 RETURN
6580 LOCATE20,14:INPUT"コリ " :F6=F6+VAL(F6$):IF F6%>FFF% THEN F9=1
6590 RETURN
6600 BEEP:PAINT(400,87),2,7:SYMBOL(350,82),"DT",1,1,7:COLOR7:LOCATE20,16:PRINT"ワ
ンツ" I9:STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6610 BEEP:PAINT(400,87),0,7,2:PAINT(400,87),6,7,2:COLOR7:LOCATE20,14:PRINT"コリ カ
" I9:STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6620 COLOR7:LOCATE20,16:PRINT"DDT ノコリ " :RETURN
6630 RETURN
6640 COLOR7:LOCATE20,14:INPUT"コリ " :A$=A$+VAL(A$):RETURN
6650 COLOR7:LOCATE20,14:INPUT"コリ " :A$=A$+VAL(A$):RETURN
6660 COLOR7:LOCATE20,15:STT=&HC0:GOSUB40010:BEEP:PRINT N$(NN):"エラ" :LOCATE20,16:P
RINT"アメル" I9:STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル"
6670 COLOR7:LOCATE20,17:COLOR7:BEEP:PRINT" キンチョリン フンシヤ " :GOSUB92
60:STT=&HC0:TEM=2:ARE=30:GOSUB40010:RETURN
6680 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6690 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6700 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6710 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6720 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6730 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6740 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6750 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6760 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6770 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6780 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6790 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6800 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6810 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6820 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6830 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6840 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6850 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6860 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6870 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6880 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6890 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6900 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6910 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6920 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6930 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6940 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6950 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6960 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6970 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6980 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
6990 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
7000 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
7010 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
7020 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
7030 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
7040 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
7050 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
7060 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
7070 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
7080 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
7090 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
7100 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
7110 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
7120 STT=&HC0:TEM=2:ARE=30:GOSUB40010:LOCATE20,23:COLOR7:PRINT N$(NN):"アメル" キチ
7130
```

```

7100 LOCATE 20,21:PRINT "RD-U-?":LOCATE 22,23:PRINT "Hit Any Key"
7110 MATI%="INKEY%":IF MATI%="" THEN 7120 ELSE PRINT "
7120 IF STAKA%=10 THEN PUT$(272,84)-(408,144),ZX,PSET,5:STT=0:TEM=5:ARE=20:GOSUB
40010:FAINT(272,90):5:PUT$(272,84)-(408,144),ZX,PSET,2:SYMBOL(0,40):" 二ホ フカツ"
,5,5,2:SYMBOL(150,150),"???" 1?4 1!":3,3,2:STT=0:ARE=30:TEM=5:GOSUB40010:CLS:TEM
2000
7123 IF TEN%=2000 THEN CLASS%=" ???? ???? ???? ???? ????":BEEP
7124 IF TEN%=1999 AND TEN%=1600 THEN CLASS%=" ???? ???? ???? ???? ????":BEEP
7126 IF TEN%=1599 AND TEN%=1400 THEN CLASS%=" ???? ???? ???? ???? ????":BEEP
7128 IF TEN%=1399 AND TEN%=1100 THEN CLASS%=" ???? ???? ???? ???? ????":BEEP
7130 IF TEN%=1099 AND TEN%=800 THEN CLASS%="????????????????????":BEEP
7132 IF TEN%=799 AND TEN%=500 THEN CLASS%="????????????????????":BEEP
7134 IF TEN%=499 AND TEN%=200 THEN CLASS%="????????????????????":BEEP
7136 IF TEN%=199 THEN CLASS%="????????????????????":BEEP
7137 SYMBOL(10,160),CLASS%,3,3,7:TEM1:=ARE=30:STT=0:GOSUB40010
7139 FOR MATI=1 TO 3000: NEXT: PUT$(372,84)-(508,144),ZX,PSET,4:PAINT(305,100),2,7,4
7150 WIDTH40,25:COLOR7,0:CLS:LOCATE10,12:INPUT "?????" 1?4 1!":A$:IF A$="" THEN
LOCATE 10,14:PRINT "?????" 1?4 1!":CLEAR:GOTO 1000 ELSE END
7500 DATA 10?4,103,15,7?4?4,130,10,7?4?4,0,124,23,7?4?4,118,37,7?4?4,115,57
7510 DATA 10?4?4,102,76,7?4?4,7,89,62,7?4?4,74,74,7?4?4,64,64
7520 DATA 7?4?4,42,72,7?4?4,41,87
7530 DATA 7?4?4,138,17,7?4?4,146,26,7?4?4,132,32,7?4?4,117,41,7?4?4,99,57,7?4?4,114,58,7
7540 DATA 7?4?4,103,78,7?4?4,95,77,7?4?4,84,74,7?4?4,99,92,7?4?4,94,70,7?4?4,83,65
7550 DATA 7?4?4,74,32,7?4?4,63,69,7?4?4,49,69,7?4?4,62,79
7560 DATA 7?4?4,47,80,7?4?4,40,76,7?4?4,39,95
7570 DATA 590,560,520,160,120,60,50,20,180,300,550,400,520,580,180,580
7580 DATA 40,38,45,120,100,110,140,150,135,150,140,100,60,40,135,40
8000 WIDTH40,25:COLOR7,1:CLS
8010 CONNECT(10,87)-(32,77)-(247,77)-(265,87)-(265,187),0
8020 CONNECT(10,84)-(30,74)-(250,74)-(270,84)-(273,184)-(250,194)-(30,194)-(10,1
84)-(10,84),0
8030 LINE(0,67)-(280,199),PSET,0,B
8040 REN=3:PC=0:RX=15:RY=76:GOSUB9090:RX=265:GOSUB9090:RY=192:GOSUB9090:RY=15:G
OSUB9090
8050 PAINT(3,70),5,0
8060 CONNECT(10,84)-(30,74)-(250,74)-(270,84)-(270,184)-(265,187),0
8070 LINE(0,67)-(280,199),PSET,0,B:PAINT(140,100),0,0
8090 LINE(0,0)-(420,65),PSET,5,BF
8100 LINE(440,0)-(629,65),PSET,5,BF
8110 LINE(15,3)-(110,14),PSET,1,BF:SYMBOL(18,5),"?????",2,1,7
8120 LINE(127,3)-(206,14),PSET,1,BF:SYMBOL(130,5),"?????",2,1,7
8130 LINE(225,3)-(302,14),PSET,1,BF:SYMBOL(226,5),"?????",2,1,7
8140 LINE(15,17)-(110,40),PSET,1,BF:CONNECT(15,17)-(111,70)-(111,40),0
8150 LINE(127,17)-(206,40),PSET,1,BF:CONNECT(127,17)-(207,3)-(207,40),0
8160 LINE(225,17)-(302,40),PSET,1,BF:CONNECT(225,17)-(303,3)-(303,40),0
8170 LINE(15,54)-(302,63),PSET,1,BF:SYMBOL(17,56),"?????",1,1,7
8180 LINE(322,54)-(420,63),PSET,1,BF:SYMBOL(324,56),"?????",1,1,7
8190 CONNECT(15,54)-(30,54)-(302,63),0
8200 LINE(450,3)-(629,63),PSET,0,BF
8210 LINE(322,43)-(420,52),PSET,1,BF:SYMBOL(324,45),"?????",1,1,7
8230 LINE(15,43)-(302,52),PSET,1,BF:SYMBOL(17,45),"?????",1,1,7
8240 CONNECT(15,43)-(30,43)-(302,52),0
8250 LINE(322,3)-(420,40),PSET,1,BF
8260 LINE(114,45)-(298,50),PSET,7,B:LINE(114,56)-(298,61),PSET,7,B
8265 CIRCLE(370,17),30,0,0,0,0: CIRCLE(370,17),4,2,2,6,0,0,0
8270 PAINT(116,47),4,7:PAINT(116,57),2,7
8280 LINE(290,100)-(641,199),PSET,0,B:CONNECT(311,103)-(293,120)-(298,179)-(31
1,196)-(625,196)-(631,179)-(631,120)-(625,103)-(311,103):PAINT(300,107),5,0
8290 CONNECT(308,106)-(622,106)-(628,120)-(628,179)-(622,196):PAINT(350,105),
0,0:PAINT(628,140),0,0
8300 PAINT(300,70),0,0,5
8310 GOSUB 9150
8315 GOSUB 9200
8320 GOSUB 10000
8330 GOSUB 10510
8335 TKTEN%=RIGHT$(" "+STR$(0,5):SYMBOL(545,85),TKTEN%,2,1,1:RETURN
8500 SPX=58180/2500:LINE(SPX+115,46)-(295,49),PSET,0,BF:LINE(67,44)-(111,51),PSE
T,0,BF:SPD%=(SPD%+SPD%)/2:RIGHT$(" "+STR$(5),5):SYMBOL(68,45),SPD%,1,1,7
9520 FUF=FX/20000*180:LINE(67,55)-(111,62),PSET,0,BF:LINE(FUX+115,57)-(295,60),P
SET,0,BF:FUL%=(FUL%+FUL%)/2:RIGHT$(" "+STR$(F,5):SYMBOL(67,55),FUL%,1,1,7
9530 RMC=MXHC/RMV:MYVC=CZ/360*6.28-1.57:MXHC=370+20*COS(CZ):KECOR%=STR$(C):OLC
%="ECOR%:MYVC=31.74-20*SIN(CZ)*.4495
9540 LINE(375,55)-(417,62),PSET,0,BF:DSP%=RIGHT$(" "+STR$(FL),5):SYMBOL(377,
56),DSP%,1,1,7
9550 LINE(375,44)-(417,51),PSET,0,BF:ALT%=RIGHT$(" "+STR$(A),5):SYMBOL(377,4
5),ALT%,1,1,7
9560 COR%=(COR%+STR$(INT(C))+". "+RIGHT$(STR$(INT(C*10)),1):COR%=(COR%+STR$(
5):LINE(330,32)-(397,38),PSET,0,BF:SYMBOL(350,32),COR%,5,1,7:ANG02%=RIGHT$("
 "+STR$(FFZ),5)
8566 IF VFGL<30 THEN LINE(370,17)-(RMC,RMV),PSET,0:CIRCLE(370,17),4,2,6,0,0,F
8570 LINE(370,17)-(MXHC,MYVC),PSET,2:VFGL=1
8572 GOSUB5700:GOSUB5500
8574 PC=7:RX=X/25+47:RY=Y*.6/25+85
8576 IF RX<24 OR RX>255 THEN RX=637
8578 IF RY<78 OR RY>184 THEN RY=137
8579 GOSUB 9095
8580 RETURN
8590 IF TAM<35 AND TAM>=1 THEN GOTO 8630
8600 IF TAM<90 AND TAM>=36 THEN KEISU=TAM:KAK=INT(RND(1)*360+1):GOTO 8640
8610 IF TAM<150 AND TAM>=91 THEN KEISU=TAM:KAK=INT(RND(1)*360+1):GOTO 8660
8620 IF TAM<220 AND TAM>=151 THEN KEISU=TAM:KAK=INT(RND(1)*360+1):GOTO8680
8630 STT=&H0C:TEM1:=ARE=15:BEEP:RETURN
8640 PY01=538-INT(KEISU*COS(KAK)/3.35):PY01=35-INT(KEISU*SIN(KAK)/3.35*.4495)
8650 XOTD1=PX01:YOTD1=PY01:PC=7:GOTO 8700
8660 PY02=538-INT(KEISU*COS(KAK)/3.9):PY02=35-INT(KEISU*SIN(KAK)/3.9*.4495)
8670 XOTD2=PX02:YOTD2=PY02:PC=6:GOTO 8710
8680 PY03=538-INT(KEISU*COS(KAK)/4.6):PY03=35-INT(KEISU*SIN(KAK)/4.6*.4495)
8690 XOTD3=PX03:YOTD3=PY03:PC=2:GOTO 8720
8700 RX=PX01:RY=PY01:GOTO 9110

```



## BASICプログラム・リスト

```

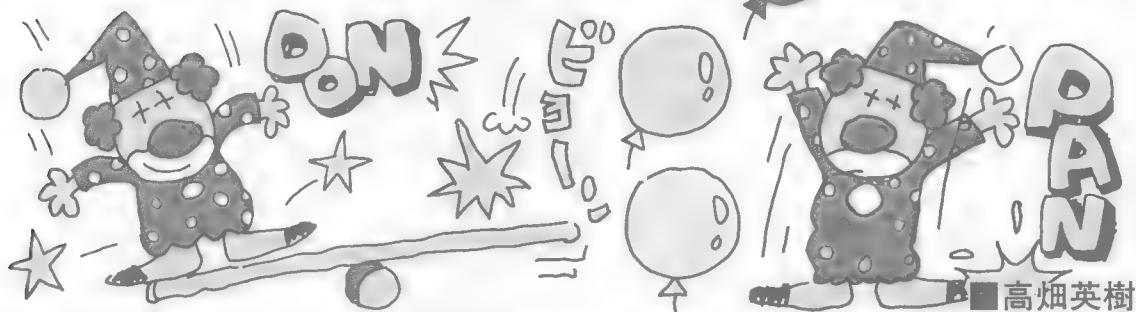
8710 RX=PX02:RY=PY02:GOTO 9110
8720 RX=PX03:RY=PY03:GOTO 9110
9000 FOR NI=1 TO 10:V=V+10
9010 REN=2:PC=6:PC1=2:RX=T(NI,1)/25+147:RY=T(NI,2)*.6/25+85
9020 GOSUB 9090:LINE(250,60)-(400,85),PSET,0,BF:SYMBOL(250,60),N$(NI),2,2,4
9030 SYMBOL(0,50+V),N$(NI),2,1,2:NEXT:RETURN
9040 FOR NI=11 TO 30:V=V+9
9050 REN=2:PC=1:PC1=6:RX=T(NI,1)/25+347:RY=T(NI,2)*.6/25+85
9055 GOSUB9090:LINE(450,60)-(580,85),PSET,0,BF:SYMBOL(450,60),N$(NI),2,2,5
9058 SYMBOL(590,V),N$(NI),1,1,6:NEXT:RETURN
9090 CIRCLE(RX,RY),REN,PC1,.6,,F:RETURN
9095 BEEP:PSET(RX,RY,PC):RETURN
9100 CIRCLE(RX,RY),REN,PC,.6:RETURN
9110 CIRCLE(RX,RY),S,PC,.6:STT=%HC0:TEM=3:ARE=35:GOSUB40010
9120 CIRCLE(RX,RY),S,0,.6
9130 RETURN
9150 FOR III=0TO6.28 STEP 3.14/6
9151 FOR AZU=15 TO 45 STEP 15
9159 XEN=540+AZU*COS(III):YEN=33-AZU*SIN(III)*.52:PSET(XEN,YEN,7)
9160 NEXT AZU:NEXT III
9197 SYMBOL(531,20),"+",2,1,4:SYMBOL(470,22),"r",2,1,4:SYMBOL(600,20),"r",1,2,4,
3:SYMBOL(592,45),"",2,1,4:SYMBOL(490,46),"",1,2,4,3:LINE(455,5)-(624,61),PSET,
2,8:LINE(460,7)-(619,59),PSET,6,B
9198 RETURN
9210 PC=0:FOR XZUB=1TO30 STEP3:PC=PC+1:LINE(470+XZUB,10+XZUB)-(606-XZUB,55-XZUB)
,PSET,PC,B:IF PC>6 THEN PC=1
9220 NEXT XZUB
9230 FOR XZUB=1TO46STEP2.1:LINE(470+XZUB,10+XZUB)-(606-XZUB,55-XZUB),PSET,2,B:NE
XT XZUB
9240 LINE(470,10)-(606,55),PSET,0,BF:GOSUB 9150
9250 RETURN
9260 PIP=PIP+1:ON PIP GOTO 9261,9263,9265,9267,9268
9261 PAINT(437,90),0,7:RETURN
9263 PAINT(452,90),0,7:RETURN
9265 PAINT(471,90),0,7:RETURN
9267 PAINT(490,90),0,7:RETURN
9268 PAINT(509,90),0,7:RETURN
9270 PIP=0:PUT(72,84)-(208,144),ZX,PSET,4
9270 FOR NI=1 TO 10
9280 REN=2:PC=2:PC1=2:RX=T(NI,1)/25+47:RY=T(NI,2)*.6/25+85
9285 GOSUB 9090
9290 NEXT
9290 RETURN
10000 PIP=0:FOR XZUB=1TO3.6 STEP1.2:LINE(296-XZUB,73-XZUB)-(634+XZUB,95+XZUB),PS
ET,2,B:NEXT XZUB
10010 CONNECT(305,85)-(356,80)-(386,81)-(406,78)-(406,91)-(386,89)-(356,90)-(305
,85),7:PAINT(328,85),4,7
10015 LINE(530,82)-(630,95),PSET,5,BF:SYMBOL(555,74),"SCORE",1,1,7:SYMBOL(350,82
),"DT",1,1,2
10020 FOR XZUB=1TO95 STEP 19:CONNECT(425+XZUB,95)-(439+XZUB,95)-(435+XZUB,85)-(4
35+XZUB,83)-(432+XZUB,75)-(429+XZUB,83)-(429+XZUB,85)-(425+XZUB,95),7:PAINT(427+
XZUB,93),6,7:PSET(432+XZUB,90,2):PSET(432+XZUB,91,2):PSET(432+XZUB,92,2):NEXT XZ
UB:RETURN
10510 GOSUB 5700:GOSUB 5500
10512 IF NAV%<N$(NN) THEN LINE(16,16)-(111,39),PSET,1,BF
10520 NAV%=N$(NN):SYMBOL(15,18),NAV%,2,2,5,4
10530 OKM%=STR$(INT(RA))
10540 IF KM%>OKM% THEN LINE(128,16)-(207,39),PSET,1,BF
10550 KM%=RIGHT$(" "+OKM%,5):SYMBOL(127,18),KM%,2,2,5,6
10560 OTARG%=STR$(INT(AN%10)/10)
10570 IF TARG%<OTARG% THEN LINE(224,16)-(303,39),PSET,1,BF
10580 TARG%=RIGHT$(" "+OTARG%,5):SYMBOL(223,18),TARG%,2,2,5,6
10590 RETURN
30000 RESTORE30070
30010 M=0
30020 READ A$:IF A$="Z" THEN RETURN
30030 IF A$="0" THEN READ CID:M=M+CID-1:GO TO 30060
30040 B=VAL("&H"+A$)
30050 ZX(M)=B+65536!* (B>32767)
30060 M=M+1:GO TO 30020
30070 DATA 0,49,40,0,7,1,D000,0,7,84,0,8,C180,0,7,40,4000,0,7,6018
30080 DATA 0,7,C0,386,3800,0,5,8,4000,3600,0,6,FE0,16,0,6,1E,0,1,18B
30090 DATA 0,6,1110,0,1,5000,0,5,53,5E00,3DF0,0,6,590,C061,0,6,2,2010
30100 DATA 4000,0,6,378,740,0,6,1,C800,4000,0,6,4B,0,8,7780,0,7,E7,8000
30110 DATA 0,7,4040,0,7,22,5000,0,7,600C,0,7,10,400,0,7,801,0,7,10,180
30120 DATA 0,7,800,8000,0,6,1C,C0,0,7,1001,8000,0,6,C10,100,0,6,6,7807
30130 DATA B000,0,4,200,F,C0,400,0,4,5,0,1,600,8002,0,6,7,3F80,4100,0,3
30140 DATA 200,E,0,1,1CE0,1,8000,0,2,2,0,1,18E0,30,0,1,80,0,3,200,3F0
30150 DATA FFC,2000,0,1,C000,0,3,3E,0,1,3F0,0,1,180,0,3,20,2000,0,2,8001
30160 DATA 0,4,1FFF,9FDC,E01C,0,1,280,0,3,218,739,97DF,10C0,809,4000
30170 DATA 0,2,4,1500,8144,C440,A000,A10,0,3,F,C0CF,1C1F,C09C,77A,7000
30180 DATA 0,3,5E0,3306,D80,F7ED,43C0,0,3,1,9011,C98,4080,BC1,0,4,170
30190 DATA 3898,7031,8000,800,0,4,C060,3000,F00,0,5,40,2000,0,2,200,0,4
30200 DATA 3C20,0,7,3B,E000,0,7,380,0,7,2,0,3,20,0,4,C0,0,7,6,8000,0,12,4000,Z
40010 GOSUB 40060
40050 RETURN
40060 POKE &H7E30,STT:POKE &H7E1B,TEM:POKE &H7E32,ARE:EXEC &H7E00:BEEP 0:RETURN
40070 DATA 34,36,BE,7E,30,10,BE,7E,32,A6,80,C6,0B,48,34,02
40080 DATA 25,04,86,81,20,01,4F,B7,FD,03,86,09,4A,26,FD,35
40090 DATA 02,5A,26,E9,31,3F,26,E1,35,36,39,12,12,12,12,12
40100 DATA 00,00,0F,FF,ZZ
50000 RESUME 2050

```

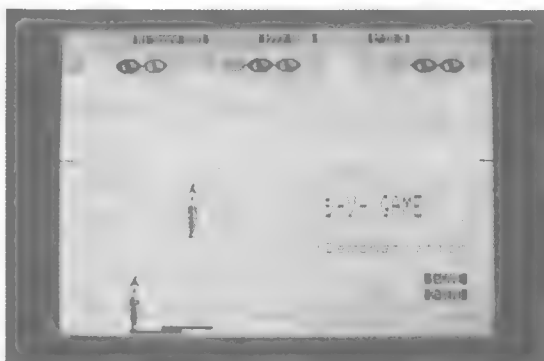
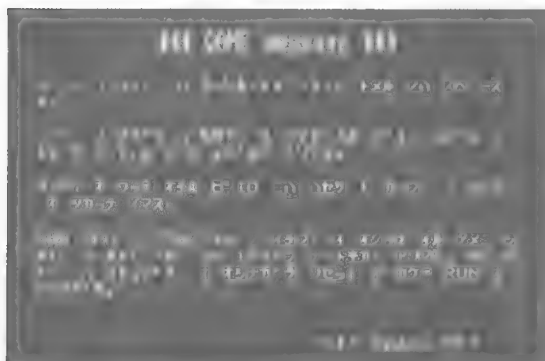


FM-8

# シーソーゲーム



■高畑英樹



FM-8のゲームができたので発表します。私なりにサブCPUをうまく使っているつもりです。FM-8でしかできない芸もあります。

## ゲームの説明

タイトルの一部に漢字が使われていますが、当方の漢字ROMよりパターン・データをもらってRAM上へ写して表示しているため、漢字ROMが未実装でも表示されます。

ピエロの動きについては簡単なパターンで複雑な動きを取り入れた方が面白いかもしれませんが、それではせっかくのFM-8の繊細な表示機能が発揮できず、つついパターンに力を入れてしまいました。

そこでゲーム性をだすため、ピエロの落とし方に乱数を使うことにしました。風船を割らずに落ちるときはまっすぐ下へ。また、割ったときはそれぞれの風船の反発力により、その落下速度とカーブをランダムに設定しています。

いま一つゲーム性を出すため、50点を追加することにシーソーの台の長さを初期状態の長さの1/6ずつ短くするようにしています。

また、画面の右端より無限の風船が隣りの風船に引っ張られるように常時登場しています。

単純なゲームですが油断ができず、高得点を得るためにはデモンストレーションでも見て落ち方のタイミングなどをつかんだ上で再挑戦してください。

## 史上最大の特徴

ゲームの始めにはテーマ音楽（天然の美の一部）の演奏と同時に風船の初期化が行なわれてピエロも登場しますが、この場面では終始演奏と同時にアニメーションが進行して他のマイコンでは絶対まねのできないFM-8のみが有する2CPUの威力が発揮されます。

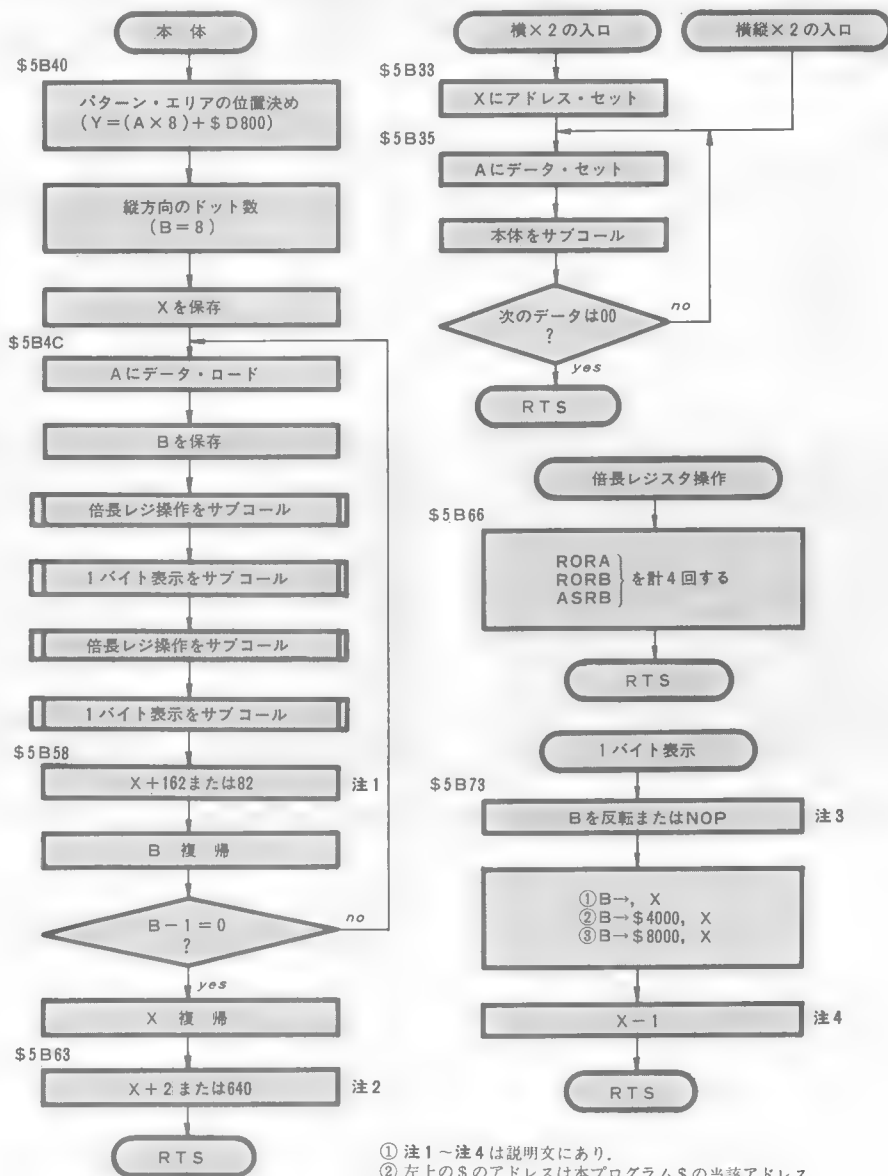
## サブCPUのプログラムについて

すでにご存知の方はこの項目は読みとばしてください。高速表示処理についてはサブCPUのRAM上で走らせるべきプログラムをメインのRAM上で作り、初期設定としてこれをサブCPUへ送った後にメインよりサブへ各種の表示コマンドを指定して実現しています。

この方法ではBIOSを通じての表示処理は不能となり、表示プログラムのすべてはサブのRAM上になければいけません。そしてサブのCPUでは特定番地(\$0~\$3E7Fは青色、\$4000~\$7E7Fは赤色、\$8000~\$BE7Fは緑色)にデータを書き込むだけで、そのデータが画面に反映されるというまったくの超スピード(CPUの処理速度そのもの)で映像処理ができます。



図1 キャラクタの倍長表示



その反面、いままで使った便利な表示命令が一切使えず、それに見合う最小限必要なプログラムから手掛ける必要があります。今回はSYMBOL文の横のみ2倍と縦横共2倍、それにドット単位の色付けPUT（ブロック）文をサブルーチンの形で紹介します。他にもっと良い方法があるかも知れませんが、私なりに精一杯考えたつもりなので承ください。

また、以降での説明文、図のアドレスはメイン上の仮のアドレスで、サブではこのアドレスへのジャンプは不能です、すべてブランチ命令を使ってください。

## サブCPUにおける キャラクタの倍長表示

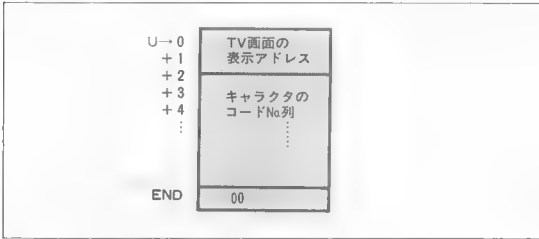
この特長はキャラクタROMのデータをレジスタ操作で倍長して表示しているため、倍長されたデータのバッファが不要で、かつ高速処理となっています。

図1にフローチャートを表します。パラメータにUレジスタを使ってデータ・テーブルのトップ・アドレスを入れます。データ・テーブルのフォーマットは図2に示しますが、データの最後は必ず00で終わってください。

また、これは横のみ2倍の手順で\$5B33が入口となりますが、縦横2倍の場合は入口が\$5B35となり、データ・フォーマットのトップはすぐキャラクタ・コードのデータから始まります。パラメータはXレジスタとUレジスタをセットします。Xレジスタは出力アドレスです。

次にサブルーチンをコールする前に注1～注4を目的に

図2 データ・テーブルのフォーマット(キャラクタ)



応じてセットする必要があります。

注1 横のみ2倍は82(10進)をセットしてサブコールします。

縦横共2倍は162(10進)をセットしてサブコールした後Xレジの内容から注2のオフセット値をマイナスして、さらにプラス80(10)かマイナス80(10)してUレジスタを前回と同じにセットして再びサブコールします。

注2 Aレジスタの内容変更のみで一定の方向へ表示していくためのXレジスタのオフセット値をセットします。2は右横へ詰めて表示していきます。また640(10進)は下へ詰めて表示してゆきます。640以上の80の倍数は上下方向へ間隔を作ります。その他の値は斜方向か間隔ができます。

注3, 注4 表示する色またはパターンの反転などをセットします。

注3をCOMBして注4の①~③をすべてストアすると白地の黒抜きの字となります。注3をNOPにすると黒地に白字となります。

注4の①~③の組み合わせで各色が表現できます。①のみストアは青、②のみストアは赤、③のみストアは緑です。

これは私の使っている方法ですが、このサブルーチン1個ですべての色、大きさを表現するにはサブコールする前にサブルーチンの命令の一部(注1~注4)を用途に応じて変更するようにしています(プログラムによる命令セット)。

## サブCPUにおけるドット単位の色表示

この特長はデータの入ったレジスタの内容で立っているビットのみ出力されて0のビットは現在表示されている画面に影響を及ぼさない点が最大の特長です。

そこですべての色のサブルーチンを作っておくと簡単にドット単位の色付け処理ができます。また、ドット・データ作成にも有利になります。図3に赤色の出力例のフローチャートを示します。本プログラム中で赤は\$5657、緑は\$5681、青は\$56AB、黄は\$56D5、ピンクは\$56FC、そして黒は\$5723がそれぞれの入口番地となっています。参考にしてください。

パラメータとしてXレジスタにTV画面の表示アドレスを、そしてYレジスタにフリーエリア2バイト分のトップ・アドレスを、そしてUレジスタにはパターン・データ・テーブルのトップ・アドレスを入れてサブコールします。データ・テーブルのフォーマットは図4に示します。フォーマットの注1はデータを横方向かまたは縦方向に組み立てるかを決めます。1のときは横方向(右へ)に1列のドットで8ドットずつ表示していきます。また80(10進)は縦にレジスタ(8ドット)の幅で下へ表示していきます。その他の数値は斜めか間隔ができます。

図3 ドット単位の色表示(赤の例)

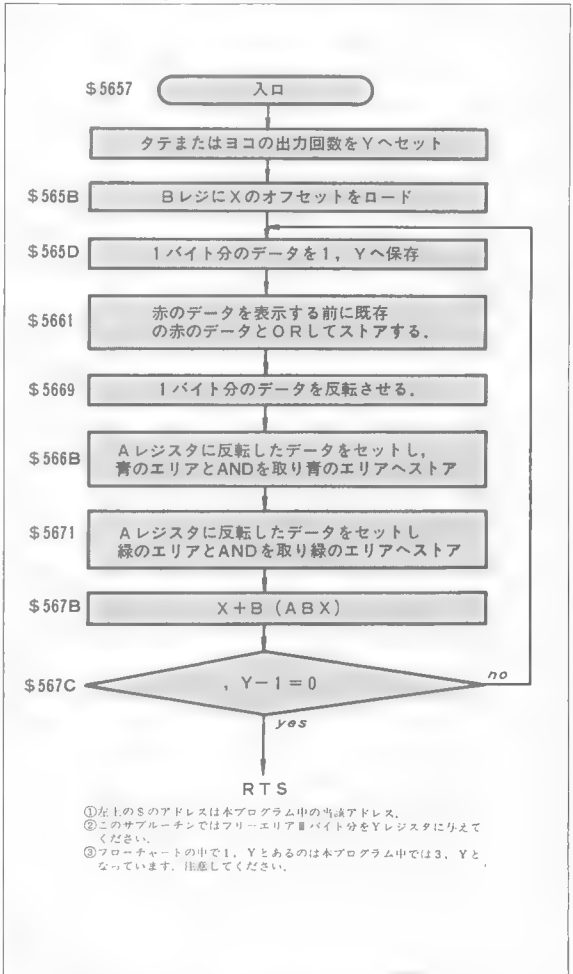
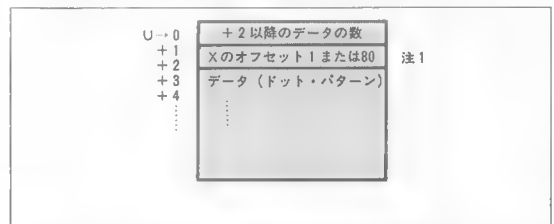


図4 データ・テーブルのフォーマット(ドット)



## 最後に

サブCPUの使用法についてはプログラム転送などソフト入りで解説したI/O別冊FM-8活用研究を参考にしてください。

また、同書の中にサブシステムのモニタ・ソース・リストがあり、この中にはCRT表示に便利なサブルーチンが豊富にあると思われますが、現時点ではそれらをサブコールするパラメータもまたデータ・フォーマットも不明です。更にコマンドにより何バイトかのバッファまたは作業エリアが要求され、我々の入るプログラムに制限を受ける可能

性もあると思われるので手離して使用できないのが現状です。

これも時間の問題で解決すると思います。一日も早い解説を、そして皆さんからの報告を期待します。

そして、これからのFM-8は高速性が要求されるゲームなどのプログラムについてはすべてサブにやらせてメインは音出し専門といった時代になりそうです。

皆さんでFM-8をゲーム・マシンとして立派に育てようではありませんか!!

## 参考文献

- 1) "サウンド・コンパイラ", I/O, '82年6月号
- 2) "SPACE INVADERS", I/O, '82年4月号

## シューターゲーム BASICプログラム・リスト

```

10 COLOR4,0:WIDTH40,20:SYMBOL(165,10),"*** GAME message ***",2,2,4:LOCATE1,3:PRINT"シューター ノ イットー ハ カースル イットーキー。 トメル ノハ スペース キー。"
20 LOCATE1,6:PRINT"シューター ノ ナカサハ 50テン オ ツイカスル コトニ ハシメ ノ ナカサノ 1/6 ツツ ミシカク ナリマス。"
30 LOCATE1,9:PRINT"フセン オ ワツテ オチル ヒ°ロ ニハ ハヤサト カーフ"ニ ランスー オ リカッチ イマス。"
40 LOCATE1,12:PRINT"SUB CPU ニ フ°ロ°ラム ノ イチフ° オ オクツテ イル タメニ ショト"ノ キットー カ フシユーフ "シ ナトキカ"アリマス。 サイ キットー ハ ストップ°キー オ オシナカラ リセツト オ カクテ RUN オ シテクダサイ"
50 LOCATE24,19:PRINT"HIT Space KEY";
60 A$=INKEY$:IFA$=" " THEN70ELSE60
70 COLOR0,7:WIDTH80,20:DEFINTA-Z:DEFUSRO=&H5029:LINE(0,198)-(639,199),PSET,4,BF:LINE(0,0)-(639,18),PSET,5,BF:CONNECT(0,87)-(20,87)-(13,87)-(0,107),0:CONNECT(639,87)-(619,87)-(626,87)-(639,107),0
80 EXEC&H5000:I=23:I=USRO(I)
90 I=21:I=USRO(I):EXEC&H5008:I=PEEK(0):IFI=49THENI=19:I=USRO(I):GOTO100ELSEIFI=50THENI=19:GOTO240ELSE90
100 I=5:I=USRO(I):SW=3
110 I=13:I=USRO(I):I=25:I=USRO(I):I=RND:IFI>.5THENI=1:J=7ELSEI=2:J=8
120 I=USRO(I):J=USRO(J):I=101+INT(RND*5):I=USRO(I):I=22:I=USRO(I):EXEC&H5D07:I=12:I=USRO(I):EXEC&H5D7B:I=11:I=USRO(I):I=26:I=USRO(I)
130 EXEC&H5021:EXEC&H5004:EXEC&H5008:EXEC&H5008:EXEC&H5008:EXEC&H5008:I=PEEK(0):IFI=0THEN130ELSEIFI=1THEN140ELSEIFI=2THEN190ELSEIFI=3THEN200
140 I=10:I=USRO(I):BEEP1:BEEP0
150 EXEC&H501D:GOSUB180:EXEC&H5008:EXEC&H5008:EXEC&H5008:EXEC&H5021:GOSUB180:EXEC&H5021:I=PEEK(0):IFI=1THEN150ELSEIFI=0THENI=103:I=USRO(I):GOTO130
160 BEEP1:BEEP0:I=101+INT(RND*4)
170 BEEP1:BEEP0:I=USRO(I):I=18:I=USRO(I):BEEP1:BEEP0:GOTO130
180 FORI=1TO70:NEXT:RETURN
190 GOSUB230:I=16:I=USRO(I):SW=SW-2:GOTO210
200 GOSUB230:I=17:I=USRO(I):SW=SW-1
210 GOSUB230:I=19:I=USRO(I):EXEC&H5DA3:FORI=1TO5000:NEXT:IFSW>0THEN110ELSEI=20:I=USRO(I)
220 I=21:I=USRO(I):EXEC&H5008:I=PEEK(0):IFI=49THENI=19:I=USRO(I):GOTO100ELSEIFI=50THENI=19:GOTO240ELSE220
230 BEEP1:BEEP0:FORI=1TO500:NEXT:RETURN
240 I=USRO(I):I=5:I=USRO(I):I=6:I=USRO(I):I=23:I=USRO(I):I=1:I=USRO(I):I=7:I=USRO(I)
250 I=14:I=USRO(I):I=101+INT(RND*5):I=USRO(I):I=12:I=USRO(I):FORI=1TO2000:NEXT:I=11:I=USRO(I)
260 EXEC&H5021:EXEC&H5004:EXEC&H5008:EXEC&H5008:EXEC&H5008:EXEC&H5008:I=PEEK(0):IFI=0THEN260ELSEIFI=1THEN270ELSEIFI=2THEN310ELSEIFI=3THEN320
270 I=10:I=USRO(I):BEEP1:BEEP0
280 EXEC&H501D:GOSUB180:EXEC&H5008:EXEC&H5008:EXEC&H5008:EXEC&H5021:GOSUB180:EXEC&H5021:I=PEEK(0):IFI=1THEN280ELSEIFI=0THENI=103:I=USRO(I):GOTO260
290 BEEP1:BEEP0:I=101+INT(RND*5)
300 BEEP1:BEEP0:I=USRO(I):I=18:I=USRO(I):BEEP1:BEEP0:GOTO260
310 GOSUB230:I=16:I=USRO(I):SW=SW-2:GOTO330
320 GOSUB230:I=17:I=USRO(I):SW=SW-1
330 GOSUB230
340 I=21:I=USRO(I):EXEC&H5008:I=PEEK(0):IFI=49THENI=19:I=USRO(I):GOTO100ELSEIFI=50THENI=24:GOTO240ELSE340

```

## シューターゲーム マシン語ダンプ・リスト

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5000	17	00	4D	39	C6	03	20	25	17	00	32	86	04	B7	FC	80	:B1
5010	B6	FC	F7	87	00	00	7F	FD	05	7F	FD	05	39	C6	09	20	:8C
5020	0C	C6	0E	20	08	C6	0F	20	04	E6	03	20	00	17	00	0D	:2E
5030	F7	FC	80	7F	FD	05	39	7F	FD	05	7F	FD	05	B6	FD	05	:E7
5040	2B	F5	86	80	B7	FD	05	B7	FD	05	B6	FC	80	26	EB	39	:11
5050	B6	FD	05	2B	F8	1A	50	86	B0	B7	FD	05	8D	74	7F	FD	:84
5060	05	8E	00	00	30	1F	26	FC	B6	FD	05	2B	FB	86	B0	B7	:9F
5070	FD	05	17	00	91	7F	FD	05	8E	00	00	30	1F	26	FC	B6	:E0
5080	FD	05	2B	FB	33	8D	00	FD	02	30	8D	0C	40	BF	00	00	:C6
5090	FF	86	B0	B7	FD	05	B7	FD	05	A6	00	B7	FC	B1	F7	FC	:04
50A0	80	7F	FD	05	7F	FD	05	B6	FD	05	2B	FB	86	B0	B7	FD	:1A
50B0	05	B7	FD	05	B6	FC	80	26	E8	11	B3	00	00	26	D2	B6	:40
50C0	01	B7	FC	80	7F	FD	05	7F	FD	05	8E	00	00	30	1F	26	:39
50D0	FC	39	10	8E	FC	80	30	8D	00	0A	C6	21	A6	80	A7	0A	:6A
50E0	5A	26	F9	39	00	00	3F	59	41	4D	41	55	43	48	49	93	:D5
50F0	D3	BF	90	34	03	1A	50	B7	D4	0A	B6	D4	0B	B6	D4	09	:4D

Sum:	5E	A9	B0	71	21	A5	5F	C6	0A	D2	5E	40	9B	65	48	FC	:D1
Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5100	B6	D4	0A	35	83	00	10	BE	FC	80	30	8D	00	0A	C6	41	:34
5110	A6	80	A7	A0	5A	26	F9	39	00	00	3F	59	41	4D	41	55	:DB
5120	43	48	49	91	D3	96	3E	B1	00	2C	93	3E	B1	90	CE	C0	:29
5130	00	7F	D3	80	7F	D3	80	7F	D3	80	7F	D3	80	B6	D4	0A	:DC
5140	B6	D3	80	27	F8	B1	01	27	0D	B7	D4	0A	B7	D4	0A	B6	:8E
5150	D3	B1	A7	C0	20	D8	7E	C0	00	00	7F	D3	80	10	CE	CF	:73
5160	FF	7F	D3	80	7F	D3	80	B6	D4	0A	B6	D3	80	27	F8	B7	:16
5170	D4	0A	B7	D4	0A	31	8D	07	33	81	01	26	15	86	00	8E	:3C
5180	0F	A0	AF	21	A7	2F	B6	0F	A7	24	CC	00	A0	ED	25	16	:49
5190	02	89	B1	02	10	26	00	C1	86	4E	8E	0F	EE	20	E3	EC	:53
51A0	29	83	31	B2	E1	2F	24	08	C8	03	E1	2F	10	24	00	8C	:69
51B0	EC	27	83	3C	52	E1	2F	24	5F	EB	A8	16	E1	2F	24	52	:E6
51C0	CB	05	E1	2F	24	6C	EB	A8	16	E1	2F	25	48	A6	28	B4	:EE
51D0	01	26	45	B6	01	A7	2C	6C	2B	17	06	96	E6	2F	C3	31	:19

51E0	B0	ED	21	17	02	E6	17	05	63	17	05	F8	ED	29	AE	21	:34
51F0	ED	21	AF	29	17	06	78	EC	21	B3	03	20	ED	21	17	05	:5B
Sum:	BA	04	58	27	F8	B3	D5	6C	FF	60	AB	F4	97	AD	58	E5	:18

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5200	48	17	05	E0	EC	21	B3	2E	90	E7	2F	EE	21	34	40	16	:44
5210	02	34	A6	2B	84	01	26	BB	17	06	57	E6	2F	C3	32	50	:3B
5220	E0	21	17	05	27	17	05	BC	B6	03	A7	2C	17	08	53	16	:00
5230	FF	28	17	06	30	E6	2F	C3	30	00	20	E4	17	06	33	E6	:B3
5240	2F	C3	32	50	ED	21	17	05	03	17	05	9B	B6	02	A7	2C	:B0
5250	17	08	2F	17	08	2C	16	FF	01	B1	03	26	3C	EC	21	10	:B2
5260	B3	27	60	10	24	FF	58	17	06	08	A6	24	4D	2B	03	17	:F6
5270	06	18	EC	21	C3	00	01	E3	25	05	ED	B1	6C	2F	6A	24	:59
5280	09	17	04	08	17	05	8C	16	FE	0D	17	04	BF	17	05	54	:C2
5290	EC	E5	23	ED	25	16	FE	C1	B1	04	26	08	A6	2C	B7	14	:44
52A0	D3	F9	16	FE	B5	B1	05	26	0D	B6	32	A7	18	55	86	06	:F6
52B0	A7	AB	16	16	07	E3	B1	06	26	0F	CC	3C	5B	ED	27	CC	:64
52C0	31	BB	ED	20	16	2F	B8	16	07	7E	B1	64	25	0A	B0	64	:F5
52D0	50	3D	ED	20	16	FE	B3	B1	07	26	13	B6	C3	C6	6C	33	:AD
52E0	BD	FF	BF	A7	E2	47	E7	49	86	01	A7	44	16	FE	6C	B1	:08
52F0	10	26	00	99	B6	B3	C6	6A	20	E5	6F	2C	B6	05	A7	24	:AF
Sum:	95	9B	02	4D	BD	BC	13	1E	24	56	5F	2C	D7	FE	BD	E2	:6F

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5300	CC	00	A0	ED	25	A6	2F	B1	26	24	E9	20	CE	6F	2C	B6	:16
5310	01	A7	24	CC	00	A0	ED	25	33	BD	FF	56	B6	B1	A7	49	:56
5320	6F	44	17	05	4D	EC	21	C3	00	A0	ED	21	17	04	10	17	:E9
5330	04	17	06	FE	25	A6	AB	11	80	00	A1	2F	24	06	9B	07	:8C
5340	A1	2F	24	C9	A6	2F	BB	01	C6	11	C0	01	80	05	24	FA	:59
5350	30	BD	05	6F	3A	A6	27	B4	34	02	6A	B6	02	07	A7	C8	:59
5360	2C	6A	AB	15	26	08	B6	32	A7	AB	15	6A	AB	16	9E	06	:59
5370	45	B6	05	53	C4	0F	3D	3A	AF	AB	13	17	02	9C	AE	AB	:E2
5380	13	33	BD	06	78	35	02	17	02	B0	16	06	CF	B1	29	24	:EF
5390	21	EC	21	17	03	A7	17	04	3C	EC	24	B3	00	A0	ED	24	:16
53A0	ED	21	17	03	A7	17	04	3C	EC	24	B3	00	A0	ED	24	16	:16
53B0	FD	AB	B1	0A	26	08	0C	E0	ED	24	16	FD	9C	B1	08	5C	:5C
53C0	26	17	04	0A	AE	21	8C	0F	A0	26	00	30	B9	FB	52	50	:25
53D0	B6	02	A7	2F	AD	16	FD	B1	30	B9	FB	AE	86	4C	20	16	:86
53E0	F1	B1	0C	26	09	17	03	64	17	03	F9	16	FD	6C	B1	00	:4B
53F0	26	1A	CC	3C	6C	ED	27	CC	31	0C	ED	29	6F	2B	16	FD	:54
Sum:	63	1C	A3	14	FD	F4	9A	45	EB	60	B6	F0	7F	08	0C	23	:AD

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5400	59	CC	07	D0	B3	00	01	26	FB	16	01	05	B1	0E	10	26	:B2
5410	00	F9	B6	D4	01	B1	1C	10	26	00	B4	AE	27	8C	3C	BD	:05
5420	24	DF	EE	21	34	40	EC	29	ED	21	17	00	9F	17	04	42	:8C
5430	EC	21	C3	00	02	ED	21	ED	29	EC	27	C3	00	02	ED	27	:E2
5440	17	03	09	17	03	9E	E4	AB	16	AE	27	3A	B6	05	C6	4C	:2B
5450	6F	B9	80	00	6F	B0	6F	B9	80	00	6F	B0	6F	B9	80	00	:46
5460	6F	B0	6F	B9	80	00	6F	B0	6F	B9	80	00	6F	B0	6F	B9	:3B
5470	16	A6	2B	84	01	26	18	30	B9	FF	5A	34	BD	6E	30	B9	:AA
5480	FE	F4	E6	AB	16	3A	BD	64	35	40	EF	21	16	00	82	30	:8E
5490	89	FE	6A	3A	BD	56	30	BB	54	E6	AB	16	3A	20	E7	B1	:80
54A0	10	10	26	FF	5C	AE	27	8C	3C	53	10	25	FF	53	EE	21	:34
54B0	34	40	EC	29	ED	21	BD	14	17	03	B7	EC	21	B3	00	02	:9B
54C0	ED	21	ED	29	EC	27	B3	00	02	16	FF	72	AE	27	B6	05	:A3
54D0	A7	AE	46	C6	0C	FE	FF	B6	08	AE	B9	80	00	EF	B9	40	:5B
54E0	00	FE	B1	4A	26	F3	3A	B6	04	26	EC	39	B6	02	A7	A4	:39
54F0	C6	50	E0	AB	16	A6	AB	16	AF	B9	80	00	6F	B9	40	00	:CB
Sum:	A6	3D	07	4E	BF	10	DB	BF	B9	4A	2B	C1	A2	4C	E0	56	:54

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5500	6F	80	4A	26	F3	3A	6A	AA	26	EB	39	B1	0F	10	26	00	:AA
5510	B1	BD	03	16	FC	34	7C	80	B6	3E	B0	04	01	26	E4	AA	:A4
5520	A6	AB	10	B0	01	A7	AB	10	24	27	B6	0F	A7	AB	10	6F	:EC
5530	AB	11	A6	AB	12	B0	01	A7	AB	12	24	05	B6	04	A7	AB	:FD
5540	12	30	BD	03	8F	E6	AB	12	3A	A6	B4	33	BD	03	74	A7	:43
5550	C4	30	BD	03	6E	E6	AB	10	3A	A6	B4	A7	01	BE	06	40	:70
5560	E6	AB	11	CB	05	3A	BC	06	90	26	03	BE	06	40	17	00	:DF
5570	A9	30	BD	03	4E	E6	AB	10	3A	A6	B4	BE	06	40	E6	AB	:1B
5580	11	3A	CB	05	E7	AB	B1	4D	27	80	33	BD	04	36	17	00	:C0
5590	A9	39	B1	07	26	AB	0D	EC	21	AE	29	ED	29	AF	21	33	:33
55A0	BD	03	16	FB	B5	17	02	CA	EC	21	C3	0A	00	1F	01	33	:66
55B0	BD	04	01	17	00	CB	30	BB	4E	33	BD	03	FB	17	00	97	:E6
55C0	30	B9	FE	C1	33	BD	03	F2	17	00	00	39	B1	11	26	05	:1E
55D0	BD	03	16	FB	B5	81	02	16	29	AE	AB	13	17	00	3B	16	:89
55E0	FD	18	B1	13	26	1F	6F	2C	BE	22	60	B6	FF	A7	B9	80	:CE
55F0	00	A7	B9	40	00	A7	B0	8C	3D	ED	20	10	27	FB	5C	B6	:46
Sum:	31	93	3C	6E	F2	25	E7	51	EB	97	D9	C7	DB	77	B1	2A	:06

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5600	4A	26	FD	20	E6	B1	14	26	03	16	04	D4	B1	15	10	26	:EB
5610	05	DB	B6	D4	01	A7	2C	16	FB	40	B6	0A	A7	A4	C6	48	:7B
5620	B6	05	A7	23	B6	FF	A7	B9	80	00	A7	B9	40	00	A7	B0	:21
5630	6A	23	26	F2	3A	6A	AA	26	E7	39	31	BD	00	0C	A7	A4	:48
5640	31	BD	02	68	B6	05	34	02	17	00	00	30	B9	FC	E1	35	:C8
5650	02	4A	26	F2	39	39	A6	C0	A7	A4	E6	C0	A6	C0	A7	:73	
5660	23	AA	B9	40	A7	B9	40	00	63	23	A6	23	A4	84	A7	:24	
5670	B4	A6	23	A4	B9	00	A7	B9	80	00	3A	6A	A4	26	DD	:F5	
5680	39	A6	C0	A7	A4	E6	C0	A6	C0	A7	23	AA	B9	80	00	:1A	
5690	B9	80	00	63	23	A6	23	A4	B9	40	00	A7	B9	40	00	:DB	
56A0	23	AA	B4	A7	B4	3A	6A	AA	26	DD	39	A6	C0	A7	A4	:91	
56B0	C0	A6	C0	A7	23	AA	B4	A7	B4	63	23	A6	23	A4	B9	:05	
56C0	00	A7	B9	40	00	A6	23	A4	B9	80	00	A7	B9	80	00	:3A	
56D0	6A	26	DD	39	A6	C0	A7	A4	E6	C0	A6	C4	AA	B9	40	:7E	
56E0	00	A7	B9	40	00	A6	C4	AA	B9	80	00	A7	B9	80	00	:A6	:E3

56F0	C0	43	A4	B4	A7	B4	3A	6A	A4	26	E0	39	A6	C0	A7	A4	:BE
Sum:	EB	F2	34	B0	3D	DC	33	6E	12	4C	4B	B4	AF	24	CC	2C	:6D

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
5700	E6	C0	A6	C4	AA	B4	A7	B4	A6	C4	AA	B9	40	00	A7	B9	:76
5710	40	00	A6	C0	A4	AA	B9	80	00	A7	B9	80	00	3A	6A	AA	

# シーソーゲーム

## マシン語ダンプ・リスト

```
SBE0 63 20 32 3D 44 45 4D 4F 20 00 B1 16 26 17 CC 01 :D8
SBF0 F4 FD 3E 80 17 F9 29 FC 3E 80 B3 00 01 FD 3E 80 :E1
Sum: 4B BA 2B E3 D7 A9 AC 10 95 AA B2 EC 50 29 18 3C :C6
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
SC00 26 F2 16 F5 55 B1 17 26 48 17 FE 31 30 8D FF 68 :E8
SC10 CC E7 84 ED 84 86 40 A7 04 33 BD 00 91 17 FF 08 :8B
SC20 17 FF 08 12 12 12 12 12 12 12 12 12 12 12 12 :08
SC30 12 12 12 12 12 12 12 12 12 12 12 12 12 12 :20
SC40 12 12 12 12 12 12 30 8D FF 2E B6 80 A7 04 16 :12
SC50 09 B1 18 26 29 BE 31 B0 CE FF F6 86 27 A7 A4 B6 :AA
SC60 21 EF 89 80 00 EF 89 40 00 EF B1 C6 46 5A 26 FD :CA
SC70 4A 26 EE 30 0E 6A A4 26 E6 6F 2C 16 F4 DC B1 19 :D1
SC80 26 03 16 FF 87 B1 1A 10 26 F4 CF 8E 22 91 CE FF :6F
SC90 FF 86 10 A7 A4 B6 0A EF 89 80 00 EF 89 40 00 EF :0F
SCA0 B1 4A 26 F3 30 88 3C 6A A4 26 EA 16 F4 AC 22 92 :60
SCB0 BC 80 BF 80 20 47 41 4D 45 00 22 E2 BC B0 BF B0 :F4
SCC0 20 47 41 4D 45 00 00 00 00 00 00 00 00 00 :3A
SCD0 EC B1 1F 02 EC B4 27 11 A6 BD 00 2A B8 B1 A7 8D :D0
SCE0 00 24 B7 FD 03 EC B4 20 03 CC 01 00 B3 00 01 26 :E5
SCF0 FB 1F 20 B3 00 01 1F 02 26 DA EC B1 EC B1 26 D2 :B1
Sum: 0A 20 97 06 F5 6B 74 7D BA C6 A9 57 3F D8 00 DD :5C
```

```
Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
```

```
5D00 B6 01 B7 FD 03 09 00 30 8C 06 BD 5C D0 16 00 6A :A2
5D10 01 D0 00 91 00 02 07 D2 00 E8 00 91 01 05 00 80 :3C
5D20 00 E8 00 91 01 05 00 80 02 BE 00 5E 01 75 00 58 :EB
5D30 02 BE 00 5E 01 5F 00 5E 02 EA 00 58 01 5F 00 5E :DE
5D40 03 B4 00 44 01 75 00 58 02 BE 00 5E 02 BE 00 5E :05
5D50 01 5F 00 5E 01 5F 00 5E 02 EA 00 58 01 5F 00 5E :7E
5D60 03 B4 00 44 02 13 00 3C 02 BE 00 5E 01 16 00 78 :F9
5D70 02 08 00 80 01 05 00 80 00 00 39 30 BC 06 BD 5C :27
5D80 D0 16 00 1E 01 16 00 78 01 05 00 80 01 5F 00 5E :D7
5D90 02 2D 00 78 01 05 00 80 01 D0 00 91 00 E8 00 91 :08
5DA0 00 00 39 30 8C 06 BD 5C D0 16 00 42 00 A4 02 34 :16
5DB0 00 03 07 21 00 52 02 34 00 03 07 21 00 29 02 34 :3D
5DC0 00 03 07 21 00 A4 02 34 00 61 01 DA 00 2D 01 F7 :66
5DD0 00 03 07 21 00 58 01 F7 00 29 02 34 00 03 07 21 :08
5DE0 00 52 02 34 00 26 02 56 01 48 02 34 00 00 39 00 :BE
5DF0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :00
Sum: 64 E7 07 40 9B 23 CB 5B 69 BC 02 9D 64 6C 02 9F :AB
```

## R A N D O M B O X

### アセンブル・リストについて

マシン語プログラムで、アセンブル・リストが載ることがあります。アセンブラを使っている人には良いのですが、アセンブラのない方がマシン語を入力するときに少々わかりにくいようなので説明します。リスト1のようなモトローラ系のアセンブル・リストでは左から、ソース・プログラム番号、メモリ番地、マシン語オブジェクト、分岐番地、ラベル、オペコード、コメントの順になっています。

ここで分岐番地ですが、アセンブルを行ないマシン語オブジェクトになったときにブランチ命令などは相対番地としてアセンブルされます。相対番地だとどのメモリ番地に分岐しているかがすぐにはわからないので相対番地をメモリ番地にしたものです。この部分はマシン語オブジェクトには含まれません。

さて、マシン語の入力方法ですが、モニタのMコマンドで行ないます。リスト1では130行の5000番地からB6, FD, 05と順に入力します(リスト2、そして次の行に移るわけですが、ここで画面に出力されているメモリ番地とアセンブル・リストのメモリ番地を比較してください。マシン語が正確に入力されていれば一致します。たとえば分岐番地までも入力してしまったりすると、この部分がずれてしまうので誤りのチェックができます。また250行、260桁の00A0, 00C9のような必要な部分を入力していなくても同様です。

次に擬似命令ですが、FCB, FDB, FCCはそれらのデータをそのまま打ち込んでください。RMBについてはRMBで指定されたデータ領域のバイト数を00にしてください(別に00にする必要はないのですが)。ENDについては入力する必要はありません。

ところでFM-7/8を買入されて間もない方の質問のうち、

付属のマニュアルをよく読めばわかることが多いようです。

FM-7では特にマニュアルが良くできていて、BASICについては無論、BIOSやSUBSYSTEM、ハード的なことまでも載っています。ですから、付属のマニュアルを熟読してください。1つの命令を1度は自分自身で実行させてみてください。(補)

#### リスト1

```
00110          *
00120          *
00130 5000 B6 FD05          * START LDA HLT
00140 5003 2B FB 5000          BMI START
00150 5005 1A 50          DRCC  #50
00160 5007 86 80          LDA  #80
00170 5009 87 FD05          STA  HLT
00180 500C BD 5D 506B          BSR  MOVE1
00190 500E 7F FD05          CLR  HLT
00200 5011 8E 0000          LDX  #0
00210 5014 30 1F          L1 LEAX  -1,X
00220 5016 26 FC 5014          L1  L1
00230 501B B6 FD05          WAIT1 LDA  HLT
00240 501B 2B FB 501B          BMI WAIT1
00250 501D 33 BD 00AD          LEAU PAT3,PCR
00260 5021 30 BD 00C9          LEAX  END,PCR
00270 5025 BF 2000          STX   #2000
00280 502B C6 FF          LDB   #FF
00290 502A 86 80          SMOVE LDA  #80
00300 502C B7 FD05          STA  HLT
00310 502F B7 FD05          STA  HLT
00320 5032 00BC          RMB   188
00330          5000          END   START
```

#### リスト2

```
*M5000
5000 00-B6
5001 00-FD
5002 00-05
5003 00-2B
5004 00-FB
5005 00-1A
5006 00-50
5007 00-B6
```

## ハード編

FM-8	HEF-09の製作	笠作貴弥	210
FM-8	PSGカードの製作	COMPAC	214
FM-7	AVDエシボ-1	柏木順二	220
FM-7/8	VDPカードの製作	森田照夫	223
FM-8	リアルタイム・キーボード	長谷 茂	236
FM-7	JU-1 STICKの製作	D.O.S.のTom	238
参考資料	TTLと接続図		240



## HERO-09の製作



ETF Laboratory 笠作貴弥

本誌で「F-BASIC EXPANDER」を発表し、FM-8をソフト面からパワーアップしましたが、今回はハード面からのパワーアップということで高速CPUユニットを製作してみました。



FM-8のメインCPUはクロック1.2288MHzで動作していることは皆さんもご承知のことと思いますが、今回のハードウェアはそれを1.75MHzまで引き上げることによって演算速度の向上を図ろうというものです。

ただこの場合、単にCPUのクロックを速くするだけでなく簡単なのですが、FM-8のメインCPUクロック(4.9152MHz) インターバル・タイマ、ビーブ音、ダイナミックRAMのリフレッシュ、RS-232C用クロックなどに使っており、安易に変更することができません。

そこで、今回CPUのクロックを変更するにあたっての基本的な考えとして、

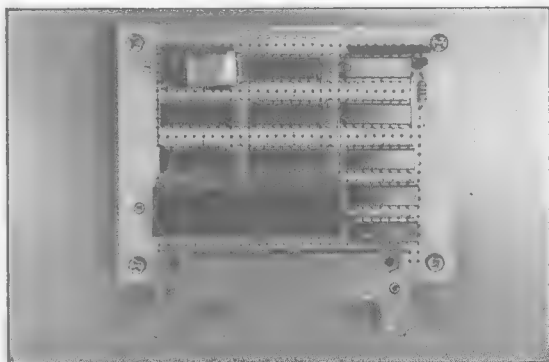
- ① いままでセーブしたテープをそのままロードできる。
- ② タイマ、I/O関係のクロックは変更しない。
- ③ 本体の改造は最小限とする(パターン・カットなどはない)。

以上を満足させるものとしてI/O関係には従来どおり4.9152MHzのクロックを供給し、CPUにはカセット関係のプログラムが動作しているときには(MOTOR ONの状態)4.9152MHz、その他のときには7MHzを供給するようにしました。

ここで、なぜ高速時CPUに68B09の最高クロックである8MHzを供給しないのかと不思議に思われる方もいると思いますが、これはFM-8に実装されているI/OデバイスやROM、RAMなどのタイミングを考えてのことです。たとえばCPUに8MHzを加えたとなると、バス・ドライバや種々のタイミング回路のデレイを考えるとROMに対するアクセス・タイムは300ns以下、DRAMに対しては230ns程度となります。

DRAMは200nsのものが実装されているので充分ですが、ROMは300nsのものが実装されており(MASK-ROM版は250ns)タイミング的に余裕がありません。また、ブートROMの2716は正確なアクセス・タイムは不明ですが(350または450nsと思われる)、一般的に2764タイプより低速なのでさらに注意が必要です。最悪の場合は2716の高速タイプ

HERO-09 試作基板



に焼きなおす必要があります。その場合2716の高速タイプより2732の高速タイプの方が入手し易いし、ほぼピン・コンパチブルなので2732に焼いた方が良いかもしれません。

以上のようなことや水晶発振子の入手性から考えて7MHzとしました。

また、CPUに8MHzを加えてROMアクセス時のみ1クロック分WAITをかける方法もありますが、この場合、部品数も多くなるしプログラム実行中にROMをアクセスする相対時間というのは結構多く、7MHz動作でWAITなしの場合よりも遅くなってしまいます。

それから③についてですが、結論から言ってパターン・カットなどはまったく必要なく、改造もメインCPUをソケットからはずして替わりに本機からのフラット・ケーブルのプラグを差すだけでOKです。ですからいつでも元の状態に戻すことができます。

## クロックの切り換え

今回一番頭を悩ませた部分で、基本的にカセットの互換性を考慮してカセットのリモートの状態(I/Oポート\$FD00のビット1の状態)によって切り換えることにしました。つまり、モータONの状態(=カセット関係のプログラムが動いている)のときは通常速度に、OFFの状態では高速に切り換えます。これは、ソフト的にもMOTORコマンドにより簡単に切り換えられて便利です。

これで切り換えの方法は良いとして、つぎは切り換えの



図1 HERO-09回路図

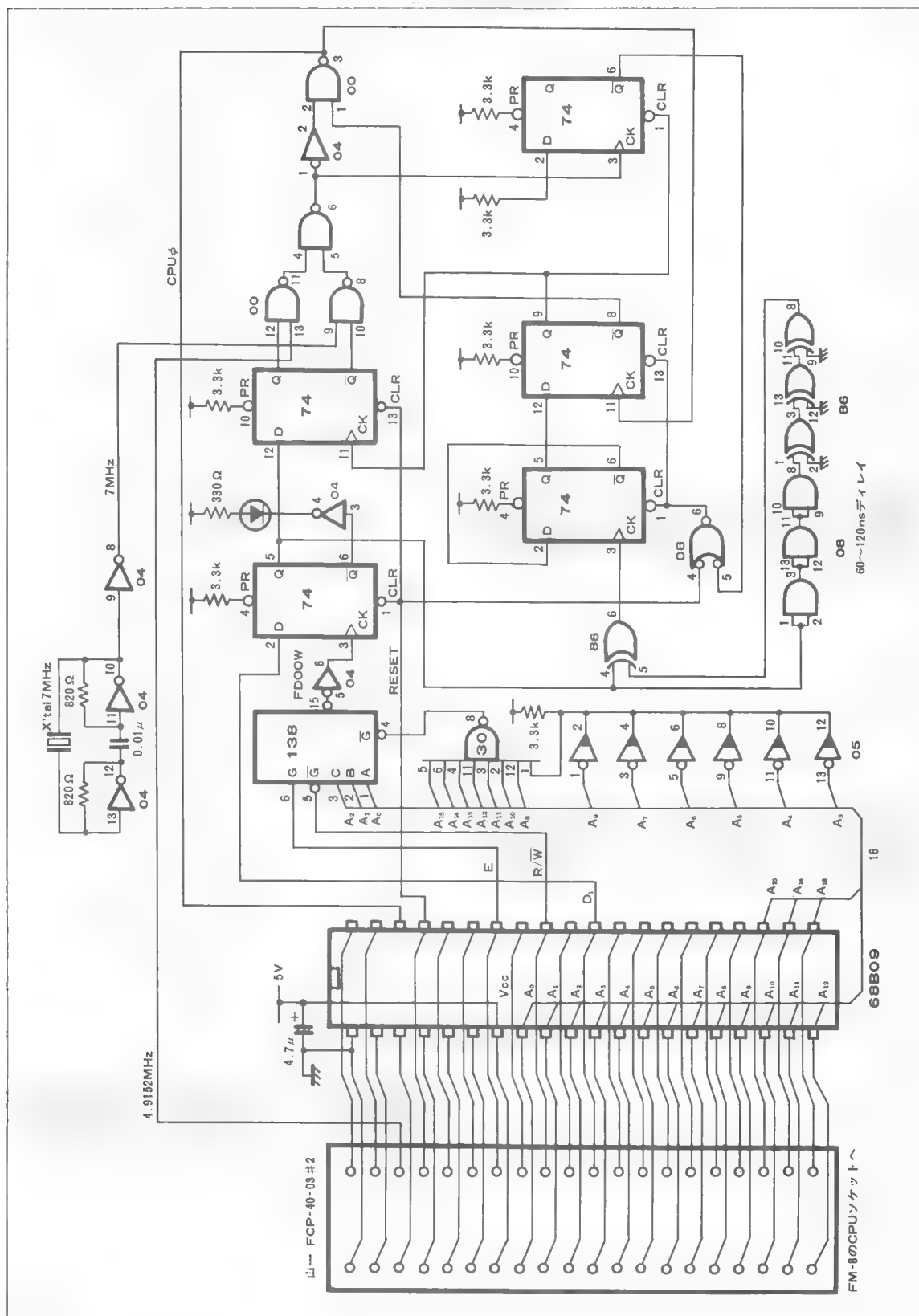


図2 ピンセットの改造方法



タイミングです。単純に\$FD00番地のビット1の内容によって切り換えていたのでは4.9152MHzと7MHzは周波数比、位相ともまったく関係ないので、CPUに変なパルスが加わって誤動作しかねません。

そこで、私は次のようにしました。始めに\$FD00のビット1が変化すると、

①現在CPUに供給しているクロックがHレベルに立ち上がった瞬間にHレベルのままにホールドする。

②クロックを切り換える。

③切り換わったクロックが立ち上がった瞬間にホールドを解除し、新しいクロックをCPUに供給する。

以上のようなロジックをフリップ・フロップを使って実現しました。つまり、位相が合うまでクロックのストレッチを行なうわけです。これによって確実に切り換えることができます。

## 回路について

回路図を図1に示します。

前項の問題を解決するためにTTL-IC11個と多少多くなっていますが、仕方ありません。CRによるタイミング回路を用いればもっと簡単にできるのですが、個人的には好きではないのでこのようにしました。

また回路図には各ICの電源ピンの配線やバスコンなどは書いてありませんが、IC2個に1個ぐらいの割り合いで、0.01~0.1μF程度のコンデンサをVccとGND間に入れてください。その他はすべて変わった点もないので、だいじょうぶだと思います。

## 部品

### ●CPU

7MHzのクロックを与えるのでBバージョン(68B09)でなければなりません。FM-8に実装されている68A09を用いても充分動作します。68A09でうまく動作しなかった場合や心配な人は68B09を買ってください。それ以外はFM-8に実装されている68A09を用いれば、¥3,000ぐらいのコストダウンです。

### ●TTL

すべて74LSタイプを使いました。

### ●X'tal (7MHz)

CPUなどを扱っているお店に行けば簡単に手に入ります。

### ●プリント基板

私はサンハヤトのICB-93Sというのを使いましたが、もう少し大きめの方が配線し易いと思います。

### ●ソケット、フラット・ケーブル

6809のソケットに差し込むプラグは山一のFCP-40-Q3#2にフラット・ケーブルを圧着し(100mm程度、実装する場所により加減する)、反対側はプリント基板に半田付けする場合FGP-40-02#1を圧着します。このとき1ピンの位置と

表1 ベンチ・マーク・テスト結果 (単位:秒)

	FM-8	FM-8 (7MHz)	PC- 8001	PC- 8801	MZ- 2000	MZ-80C Hu-BASIC	MZ-80C(4MHz) HuBASIC
プログラム1	10	6	13	18	5	11	5
2	6	4	6	11	—	12	6
3	14	9	21	23	10	38	19
4	9	6	10	11	6	9	4
合計	39	25	50	63	—	70	34

### ベンチマーク・テスト

#### テスト・プログラム1

```
10 TIME$="00:00:00"
20 FOR I=1 TO 10000
30 NEXT
40 PRINT TIME$
50 END
```

#### テスト・プログラム2

```
10 TIME$="00:00:00"
20 FOR I%=1 TO 10000
30 NEXT
40 PRINT TIME$
50 END
```

#### テスト・プログラム3

```
10 TIME$="00:00:00"
20 FOR I=1 TO 500
30 X=COS(I):Y=SIN(I)
40 NEXT
50 PRINT TIME$
60 END
```

#### テスト・プログラム4

```
10 TIME$="00:00:00"
20 I=0
30 X=I*I:I=I+1
40 IF I<1000 THEN 30
50 PRINT TIME$
60 END
```

ケーブル・マークの位置を合わせることに。

半田付けせず着脱したい場合は、ソケット・タイプFAS-40-17を圧着します。このときもFCPの1P側がソケットの△印へくるように注意しましょう。本機側にはヘッダータイプでFAP-40-07#2を取り付けます。

圧着した状態で購入したいときは次のように指定すると山一電機、または秋葉原の代理店あたりで買えます。

FCB-40-02#2, FGP-40-02#1 TYPE-1 L=100(または必要な長さをmm単位で); またはFCP-40-03#2, FAS-40-17 TYPE-1 L=100およびFAP-40-07#2

## 製作

私はいつもつぎのような順番でします。

①各部品の配置を決める。

②ICを半田付けする。

本来ならば半導体は最後にすべきなのでしょうが、先に半田付けします。半田付けに自信のない人は、ソケットを使いましょう。

③電源ラインを配線する。

0.6~1.0mm程度のスズメッキ線を使います。ループを作



# FM-8用PSGカードの製作



COMPAC

FM-8でBEEP音しかでないのはさびしいかぎりですね。  
そこで、FM-8用にPSGカードを作ってみました。

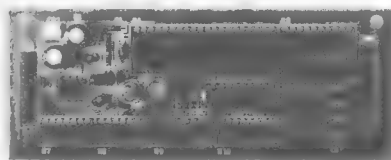
## 回路について

図1のブロック図を見てください。アドレス、EB、RWBを使ってデータ・バス、ラッチなどをコントロールしてPSGをコントロールします。コントロールは表2を参照してください。全回路図は図2です。

スピーカーからひどく雑音が出るときはPSGのVccとGNDの間に直接（Vcc側の足を短かく）0.1μFくらいの周波数特性の良いコンデンサを入れてください。

特に注意することは電源ピンです。PSGをはずした状態でショートしていないかどうか調べてください（小抵抗測定レンジ）。

LM386の6番ピンのところのコンデンサはなるべく1Cのそばに付けてください。あまり離れるとノイズがひどく



なります。

抵抗、コンデンサを付けるときにアース側、電源側は多少長くなってもかまいませんが、信号線側を短くしてください。

FM-8を壊さないために、良く配線のチェックをしてください。ボードをFMに付けてスイッチを入れて暴走した場合、急いでスイッチを切ってもう一度確認してください。特にアドレス、データ、74LS244、74LS374です。

PSGのテスト・プログラムをリスト2に示します。このプログラムでPSGの各レジスタの内容を自由に書き換えることができます。試しにR0=AE, R1=0, R7=FE, R8=Fというデータを書き込んでみてください。

図1 ブロック図

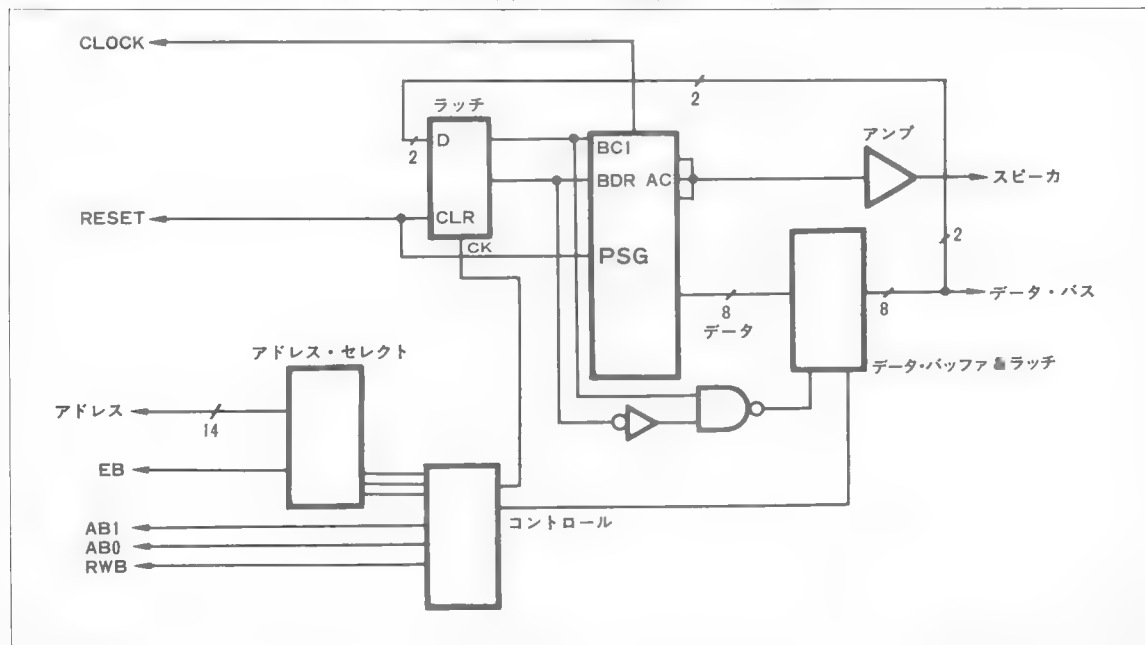


図 2

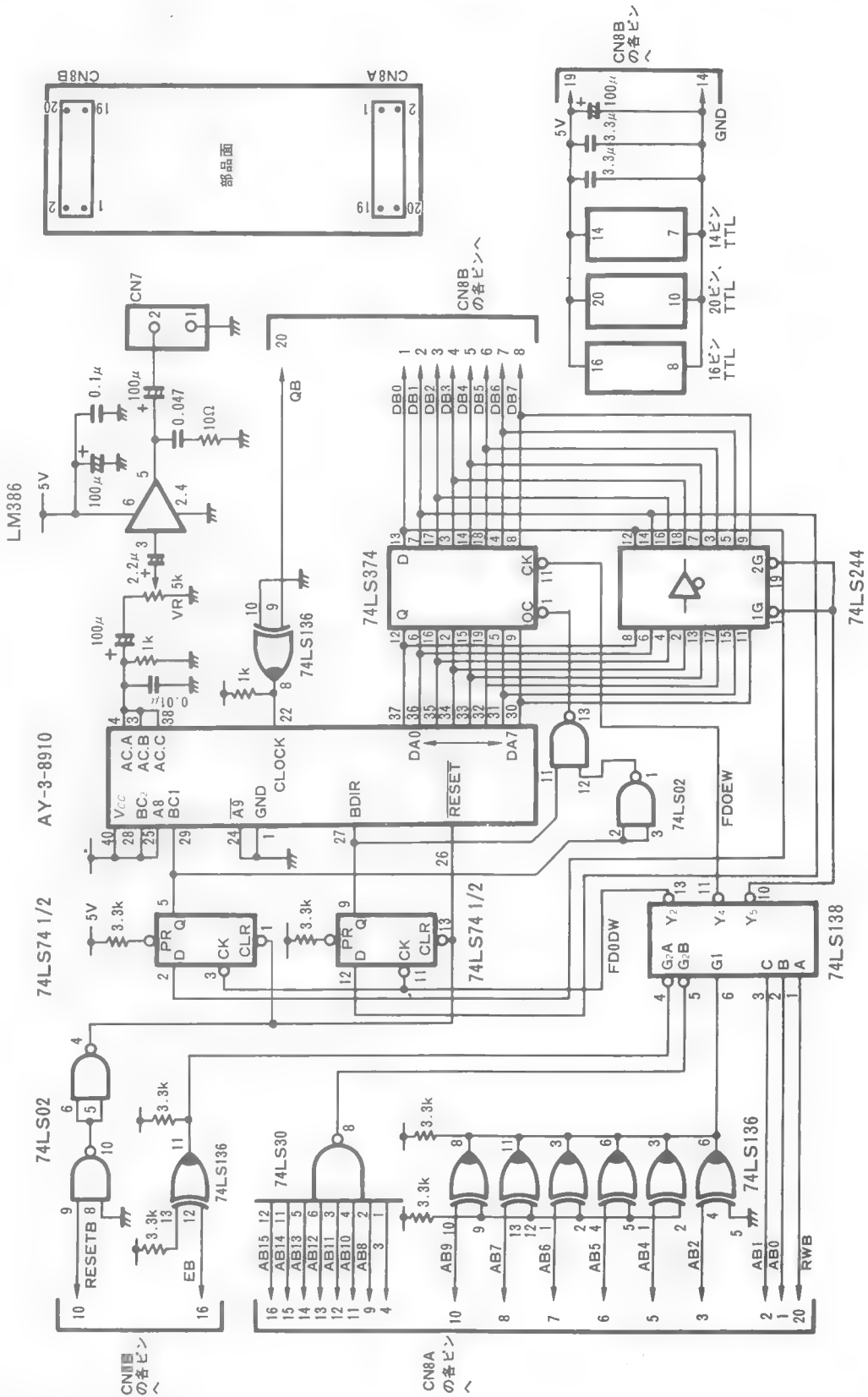


図3 基板寸法

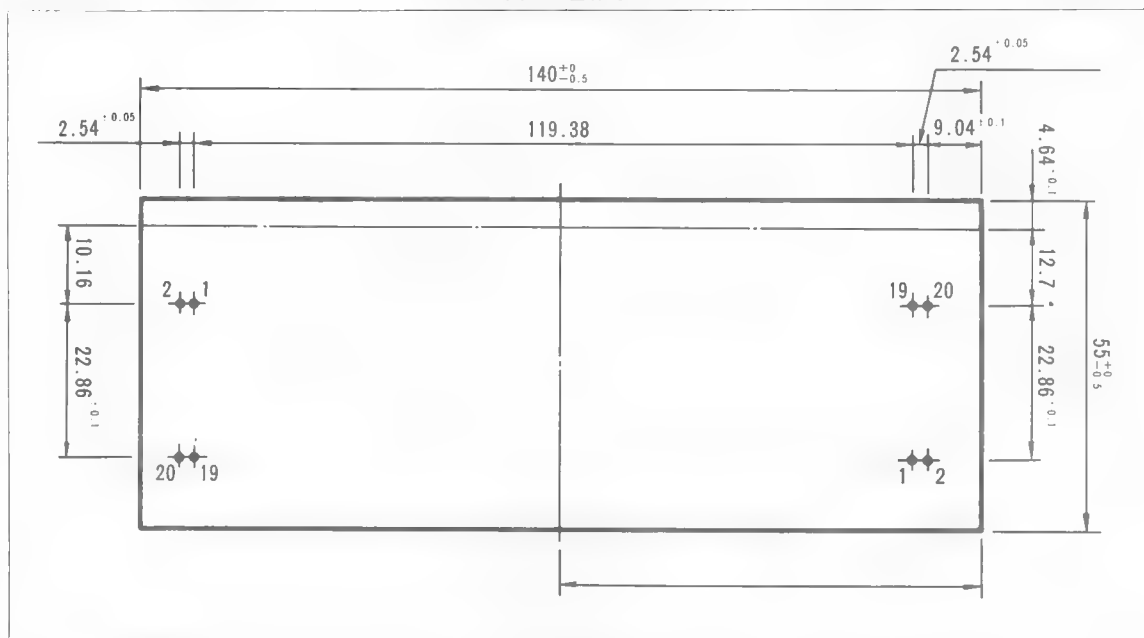


表1

TTL	74LS02	1
	74LS30	1
	74LS74	1
	74LS136	2
	74LS138	1
	74LS244	1
	74LS374	1
IC	AY-3-8910	1
	LM386N	1
抵抗	10Ω	1
	1kΩ	1
	3.3kΩ	6
	5kΩ	1
コンデンサ	0.01μ50V	1
	0.047μ50V	1
	0.1μ35V	1
	2.2μ35V	1
	3.3μ10V	2
	100μ25V	4
ICソケット	8ピン	1
	14ピン	5
	20ピン	2
	40ピン	1
コネクタ	2P	1
	(日本モレックス) 20P	2
	(日本航空電子または相当品)	

図4 PSGブロック・ダイアグラム

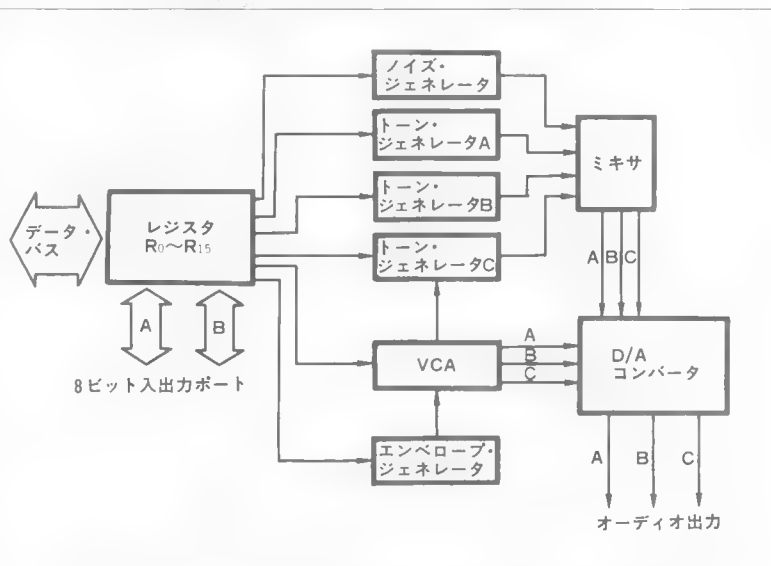


表2 BD1R, BD1, BD2のコントロール

BD2	BD1R	BD1	備考
1	0	0	インアクティブ
1	0	1	リードfrom PSG
1	1	0	ライトfrom PSG
1	1	1	ラッチ・アドレス

表3 レジスタR<sub>n</sub>とエンベロープ・パターン

レジスタR <sub>n</sub> の 下位4ビット	エンベロープ・パターン (縦軸は音量, 横軸は時間)
0 0 0 0	△
0 0 1 0	△
1 0 0 0	△
1 0 0 1	△
1 0 1 0	△
1 0 1 1	△
1 1 0 0	△
1 1 0 1	△
1 1 1 0	△
1 1 1 1	△

これで440Hzの音ができれば完成です。指定の音、またはまったく音がでないときはアドレス、74LS138、74LS74の辺りを調べてください（VRの位置はまん中くらい）。ホッ、ホッ、ホッというような音のときはコンデンサの極性を調べます。

スイッチを入れると音が出っ放しのときは、74LS02からPSGまでと74LS74の辺りを調べます。

PSGのレジスタに書き込みができないときは74LS74、74LS02、74LS138の辺りを調べます。

図5 PSGのレジスタ

レジスタ	bit	働 き	7	6	5	4	3	2	1	0
R <sub>1</sub>		チャンネルAの音階	下位 8 ビット・データ							
R <sub>2</sub>							上位 4 ビット・データ			
R <sub>3</sub>		チャンネルBの音階	下位 8 ビット・データ							
R <sub>4</sub>							上位 4 ビット・データ			
R <sub>5</sub>		チャンネルCの音階	下位 8 ビット・データ							
R <sub>6</sub>							上位 4 ビット・データ			
R <sub>7</sub>		ノ イ ズ 周 波 数	5 ビット・データ							
R <sub>8</sub>		入 出 力 の 選 択	IN/OUT ポートB   ポートA		C	ノイズ B		A	トーン C   B   A	
R <sub>9</sub>		チャンネルAの音量				M	4 ビット・データ			
R <sub>10</sub>		チャンネルBの音量				M	4 ビット・データ			
R <sub>11</sub>		チャンネルCの音量				M	4 ビット・データ			
R <sub>12</sub>		エンベロープ周期	下位 8 ビット・データ							
R <sub>13</sub>			上位 8 ビット・データ							
R <sub>14</sub>		エンベロープ波形					CONT.	ATT.	ALT.	HOLD
R <sub>15</sub>		ポート A レジスタ	8 ビット・データ							
		ポート B レジスタ	8 ビット・データ							

音階は12ビットのデータで表現する。  
SOUND DATAを参照。

ノイズの平均周波数を指定する。

各チャンネルから出す音源や、I/Oポートを指定する。

M=0のとき  
下位 4 ビットが音量調節  
M=1のとき  
エンベロープ作動

音階は12ビットのデータで表現する。  
SOUND DATAを参照。

ノイズの平均周波数を指定する。

各チャンネルから出す音源や、I/Oポートを指定する。

M=0のとき  
下位4ビットが音量調節  
M=1のとき  
エンベロープ作動

## PSG(AY-3-8910)について

PSG (Programmable Sound Generator) には、AY-3-8910の他にも、TIの76477や76489といったようなものがあります。その中でもAY-3-8910はホビュラーなもので、本誌でも何度か取り上げられている他、いろいろなパソコンに内蔵されています。

この石は、3つのトーン・ジェネレータ、ノイズ・ジェネレータ、エンベロープ・ジェネレータなどを内蔵していて、シンセサイザとしての機能（といっても簡単なものですが）を一通り備えています。読者の中にも良くご存知の人がいると思いますが、初めての人も多いはずなのでここで簡単な機能の紹介をしておきましょう。

AY-3-8910には、図5に示されているような16個のレジスタがあります。そのうちのR<sub>11</sub>とR<sub>12</sub>は8ビットのハイレベル・ポートなので、音声出力とは直接関係はありません。今回使用するのは残りのR<sub>1</sub>からR<sub>10</sub>までです。

R<sub>0</sub>からR<sub>5</sub>は、3つのトーン・ジェネレータの発振周波数を決定します。R<sub>0</sub>とR<sub>1</sub>、R<sub>2</sub>とR<sub>3</sub>、R<sub>4</sub>とR<sub>5</sub>をそれぞれ組にして使用し、そのうちの下位12ビットの値が有効です。実際の発振周波数は次のようにして計算できます。

$$f = \frac{f_{\text{clock}}}{16 \times D}$$

ここでDは、書き込むべきデータ、 $f_{\text{clock}}$ はPSGに与えるクロック。

FM-8では $f_{\text{clock}}$ の値は1.2288MHzですから、楽器のチューニングなどに使われるラの音（440Hz）を出したいのならば、\$00AEをレジスタに書き込めばよいわけです。

R<sub>6</sub>はノイズの平均周波数を決定するレジスタで、下位5ビットのみが有効です。周波数はトーンのときと同様にし、次の式で求められます。

$$f = \frac{f_{\text{clock}}}{16 \times D}$$

平均周波数を高く設定してやると、ホワイト・ノイズのような『シャー』といった感じの音になり、低く設定した場合には、ピンク・ノイズのような『ゴー』といった感じ

の音に聞こえます。実際には、PSGでヒンク・ノイズを出すのは難しいのですが、低周波数のトーンとミックスしてやることで、なんとか解決できると思います。

R<sub>7</sub>はPSGのモード設定を行なうレジスタです。対応する各ビットを0（ゼロ）にすることにより、モードを設定します。ですからジェネレータのAを、トーンとノイズのミックス出力にセットしたければ、\$F6をこのレジスタに書き込めばよいわけです。

R<sub>9</sub>からR<sub>10</sub>は音量の設定をするためのものです。下位4ビットが有効で、最大音量にセットするには\$0Fを書き込むことになります。ただし、このレジスタのビット4を1にするとエンベロープがイネーブルになり、音量の指定は無視されます。この場合、音量のコントロールはエンベロープ・ジェネレータに任せられることになります。

R<sub>11</sub>からR<sub>13</sub>はこのエンベロープのコントロールを行なうレジスタです。エンベロープの形状は表3にある中から選択できます。選択した波形のコードをR<sub>13</sub>に書き込めば、その波形がセットされ同時にそれがエンベロープのトリガとなります。エンベロープの長さは、エンベロープ周波数としてR<sub>11</sub>とR<sub>12</sub>で指定します。この周波数と波形をいろいろ変えるとトレモロがかかったような音や、ピアノのような立ち上がりの鋭い音を作ることができます。エンベロープ周波数は次式で与えられます。

$$f = \frac{f_{\text{clock}}}{256 \times D}$$

表3の下にあるεの大きさはこのfの値から、

$$\epsilon = \frac{1}{f}$$

で求めることができます。

AY-3-8910には、エンベロープ・ジェネレータが1つしかありません。ですから、3声の音楽演奏をする場合にもエンベロープはそのうちの1声にしかかけられません。むしろ、このエンベロープは波の音や爆発音などのような擬音を作りたい場合などに効果があるようです。



## PSGカードの使用方法

PSGの各レジスタは、CPUによって直接アクセスすることができません。そこで、FM-8のI/Oエリア上にコマンド・レジスタとデータ・レジスタという2つのレジスタを設けて、そこを介してPSGの各レジスタのアクセスを行なうことにします。今回のPSGカードはコマンド・レジスタに\$FD0Dを、データ・レジスタに\$FD0Eを割り当てています。

### 1) レジスタの選択

PSGの各レジスタからデータを読んだり、また書き込んだりするには前もってアクセスするレジスタの指定をしておかなければなりません。そのためには、以下のような手順をします。

- ①データ・レジスタにアクセスしたいPSGのレジスタ番号を書き込む。
- ②コマンド・レジスタにラッチアドレス・コード(\$03)を書き込む。

- ③コマンド・レジスタにインアクティブ・コード(\$00)を書き込む。

### 2) レジスタへの書き込み

PSGの各レジスタにデータを書き込みますが、その前に先の例のようにして、書き込むレジスタの指定をしなくてはなりません。

- ①データ・レジスタにPSGに送るデータを書き込む。
- ②コマンド・レジスタにライトデータ・コード(\$02)を書き込む。
- ③コマンド・レジスタにインアクティブ・コードを書き込む。

### 3) レジスタからの読み出し

まずレジスタの指定を行ってから、次のようにします。

- ①コマンド・レジスタにリードデータ・コード(\$01)を書き込む。
- ②データ・レジスタからデータを読み取る。
- ③インアクティブにする。

以上のようにして、PSGへのアクセスを行ないます。実際の例として、リスト1を参考にしてください。

リスト1

```

PAGE 001 ( ,005033)

10000 *
10010 * program PSG READ-WRITE ROUTINE
10020 * date 22-JUN-82
10030 *
10040 0000 20 04 0006 WRTENT BRA WRTPSG
10050 0002 20 19 001D REDENT BRA REDPSG
10060 *
10070 0004 FD0D PSGADR FDB $FD0D PSG address
10080 *
10090 * subroutine WRITE TO PSG
10100 * date 1-FEB-82
10110 * parameter list
10120 * A := < register_selection : 0..15 >
10130 * B := < data : 0..255 >
10140 * all registers are reserved
10150 *
10160 0006 WRTPSG EQU *
10170 *
10180 0006 34 12 PSHS A,X
10190 0008 AE 8C F9 LDH PSGADR,PCR
10200 000B A7 01 STA 1,X
10210 000D B6 03 LDA ##3 ratch address
10220 000F A7 B4 STA ,X
10230 0011 6F B4 CLR ,X inactive
10240 0013 E7 01 STB 1,X
10250 0015 B6 02 LDA ##2 write data
10260 0017 A7 B4 STA ,X
10270 0019 6F B4 CLR ,X inactive
10280 001B 35 92 PULS A,X,PC
10290 *
10300 * subroutine READ FROM PSG
10310 * date 1-FEB-82
10320 * parameter list
10330 * A := < register_selection : 0..15 >
10340 * return value
10350 * B := < data : 0..255 >
10360 * A,X,Y,U registers are reserved
10370 *
10380 001D REDPSG EQU *
10390 *
10400 001D 34 12 PSHS A,X
10410 001F AE 8C E2 LDH PSGADR,PCR
10420 0022 A7 01 STA 1,X
10430 0024 B6 03 LDA ##3 ratch address
10440 0026 A7 B4 STA ,X
10450 0028 6F B4 CLR ,X inactive
10460 002A B6 01 LDA ##1 read data
10470 002C A7 B4 STA ,X
10480 002E E6 01 LDB 1,X
10490 0030 6F B4 CLR ,X inactive
10500 0032 35 92 PULS A,X,PC

```

```

10510                                END
TOTAL ERRORS 00000--00000
TOTAL WARNINGS 00000--00000

PROGRAM BEGIN ADDR=0000
PROGRAM END   ADDR=0033
PROGRAM ENTRY ADDR=****

```

## リスト 2 PSGテスト・プログラム

```

10000  '*
10100  '*   program   PSG (AY-3-8910) CONTROL
10200  '*   date      9-AUG-82
10300  '*
10400  CLEAR 300,&H5FFF
10500  DEFINT A-Z:DEF USR0=&H6000:DEF USR1=&H6002
10600  '*
10700  RESTORE
10800  FOR ADR!=&H6000 TO &H603A:READ A$:POKE ADR!,VAL("&H"+A$)
10900  NEXT
11000  '*
11100  WIDTH 80,25:COLOR 4
11200  WHILE -1
11300  FOR I=0 TO 12 STEP 4:PRINT
11400  FOR J=I TO I+3:D=USR1(J*256)
11500  PRINT "R -":PRINT USING "###";J;
11600  PRINT "  " : "+RIGHT$("0"+HEX$(D),2)+"  ";
11700  NEXT:NEXT
11800  PRINT
11900  PRINT:INPUT "REGISTER=":R
12000  PRINT:INPUT "DATA=":D$
12100  D=VAL("&H"+D$):TMP=USR0(R*256+D)
12200  WEND
12300  END
12400  '*
12500  DATA 20,04,20,1B,FD,0D,34,46,EC,02,EE,8C,F7,A7,41,86
12600  DATA 03,A7,C4,6F,C4,E7,41,86,02,A7,C4,6F,C4,35,C6,34
12700  DATA 46,EC,02,EE,8C,DE,A7,41,86,03,A7,C4,6F,C4,86,01
12800  DATA A7,C4,E6,41,6F,C4,4F,ED,02,35,C6

```



# アナログ信号をマイコンのデータに変換

# A/Dコンバータ

■ 柏木順二

写真1 A/Dコンバータ・ボード

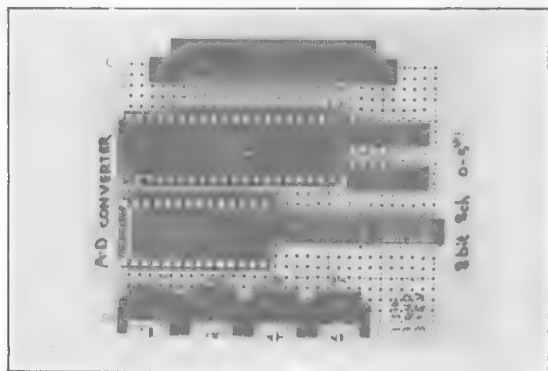


写真2 実装状態

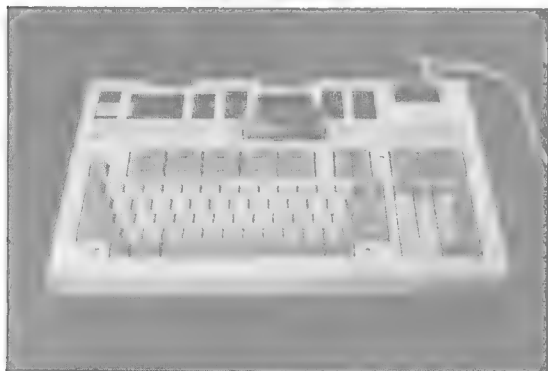
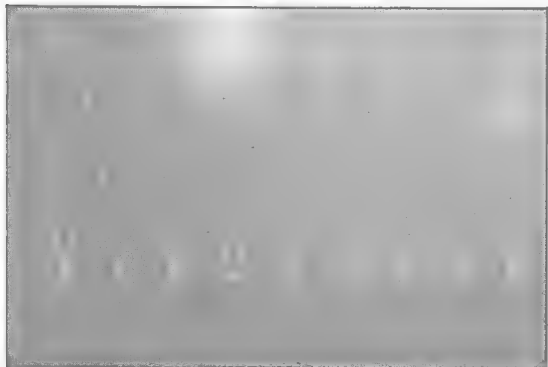


写真3 簡単な応用例



富士通のFM-7は発売されたばかりですが、FM-8のハードウェア、ソフトウェア構成を踏襲しているため内部の構造もほとんど共通でソフトウェアも互換性があります。また、システムを拡張する上で本体内にカードを実装できるというのは大きな魅力です。今回はテストとして簡単なA-Dコンバータ・ボードを製作してみました。

## オプション・スロット

本体上部の蓋をあけると3つのコネクタがあります。1つはZ80カード用の40ピン、残りはI/Oインターフェイス用の32ピンで、富士通のFCN360シリーズのコネクタが使われています。今回は32ピンの方を使うことにしました。信号線は表1のとおりです。アドレス・バスは下位8ビット(AB0~7)しか出てなく、その他にIOSがあります。これはIO領域としてFD00~FDFFが割り当てられて、上位8ビットは内部ですでにデコードされてIOSとなっているからです。またEXTDET(拡張検出)なる信号は、外部でGNDに接続しておくFD05番地のデータD<sub>0</sub>が0になります。これをソフトウェアで検出しているかどうかは不明です。CLK(2.5MHz)はRS-232C用に、4.9MHzのクロックを1/2に分周して出しているようです。

## A/Dコンバータ・ボード

以上のことをもとにして、簡単なA/Dコンバータ・ボードを設計しました。A/Dコンバータには入手しやすいADC0809(8ビット、8チャンネル)を使います。このA/Dコンバータは出力が3ステートになっているので直接バス・ラインに接続できますが、今回は一応PIAを入れてあります。STARTパルスはPIAのCB2を使ってBポートにチャンネル番号を出力したあと自動的に発生させ、EOCをフラグ検出しています。ラッチのOEは常に1です。PIAのアドレスはデコードICによってFD40~43に設定しており、簡単に変更できます。変換クロックとしては、2.5MHzを1/4に分周しています。

使用部品で重要なのは本体とのコネクタですが、型番はFCN-365 P032というものです。秋葉原ならは入手可能です(注)。基板の大きさは横124×縦78mmまでですが、蓋を取って使うならは縦はもっと長くできます。今回

は小型の基板を使いました。

# 簡単なプログラム

640点取り込んで規格化して画面に表示するだけの簡単なプログラムを作りました(リスト1, 2)。データ取り込みのマシン語サブルーチンは6000番地から、データエリアは6100番地からです。1点の取り込みにかかる時間は130 $\mu$ sec程度で、A-Dコンバータの変換時間(100 $\mu$ sec TYP.)に近い値となりました。

表1 オプション・スロット・コネクタ信号表

ピン番号	信号名	方向	ピン番号	信号名	方向
		CPU/07445			CPU/07445
A-1	AB0	→	B-1	AB1	→
2	2	→	2	3	→
3	4	→	3	5	→
4	6	→	4	7	→
5	DB0	↔	5	DB1	↔
6	2	↔	6	3	↔
7	4	↔	7	5	↔
8	6	↔	8	7	↔
9	Q	→	9	E	→
10	<u>I/O</u>	→	10	RESET	→
11	<u>R/W</u>	→	11	EXTDET	←
12	<u>TRQ</u>	←	12	CLK(2.5MHz)	→
13	+12V	→	13	-12V	→
14	+5V	→	14	+5V	→
15	GND		15	GND	
16	GND		16	GND	

# 終わりに

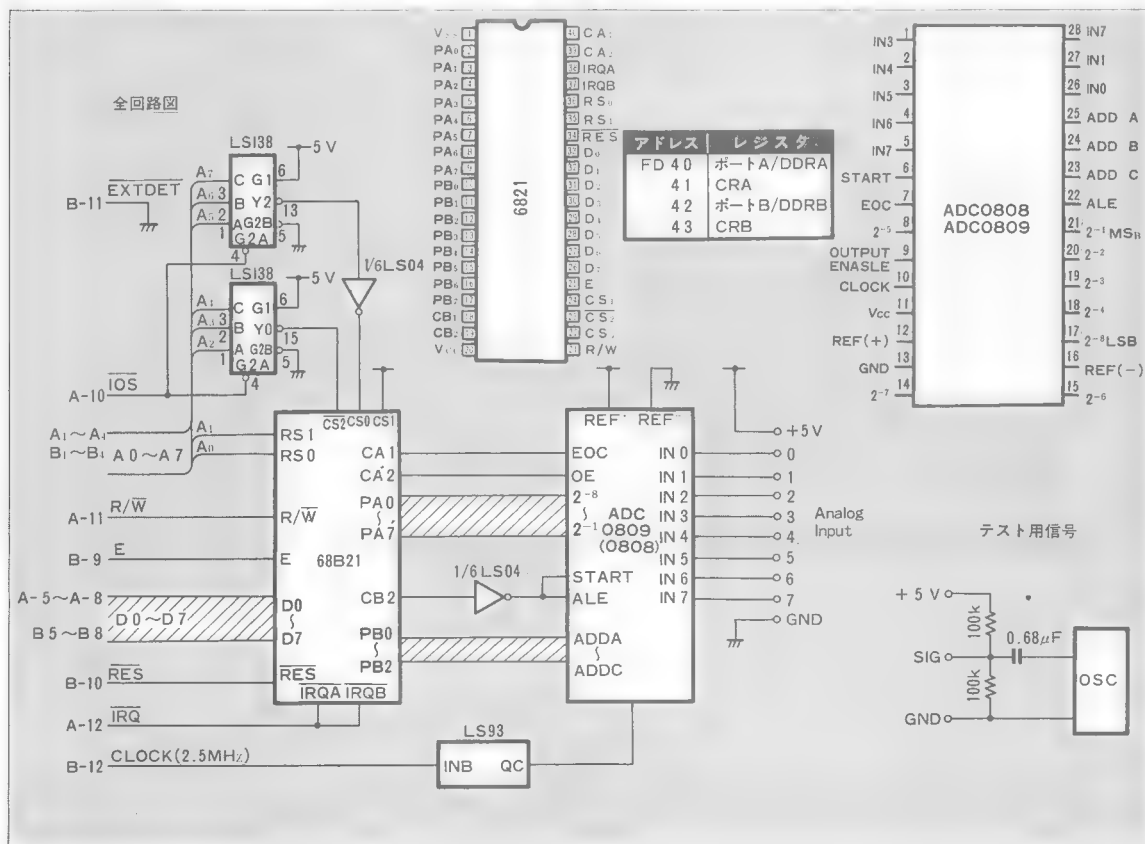
購入したばかりなのでまだ充分に使いこなしていませんが、このように簡単なインターフェイスなら1日もあればできるのでお試しください。読者の皆様のお役に立てば幸いです。

注：ラジオセンター内：第2バール無線，  
ラジオデパート内：神保商会，など。

表 2 部品表

品名	型番	個数	備考
I C	68B21	1	
	ADC0809 (0808)	1	
	74LS138	2	
	74LS04	1	
	74LS93	1	
ソケット	40ピン	1	
	28ピン	1	
コネクタ	FCN-365-PO32	1	富士通
	HNC2-2.5P-3DS	8	ヒロセムセン製
基板	ICB-93	1	サンハトヤ製
その他、配線用ビニール線		若干	

## 全回路



## リスト 1 画面表示プログラム

```

10 POKE &HFD40,&H00
20 POKE &HFD41,&H3E
30 POKE &HFD42,&HFF
40 POKE &HFD43,&H2D
50 INPUT I%
60 POKE &H6100,I%
70 EXEC &H6000
80 CLS
90 FOR J%=0 TO 639
100 A%=199-PEEK(&H6101+J%)/1.28
110 PSET (J%,A%)
120 NEXT J%
130 GOTO 50

```

## リスト 2 データ取り込みサブルーチン

```

6000 BE 6100 LDX #6100
6003 E6 80 LDB ,X+
6005 F7 FD42 STB $FD42
6008 B6 FD41 LDA $FD41
600B 4D TSTA
600C 2A FA BPL $6008
600E B6 FD40 LDA $FD40
6011 A7 80 STA ,X+
6013 8C 63B1 CMPX #63B1
6016 2B ED BMI $6005
6018 39 RTS

```

## RANDOM BOX

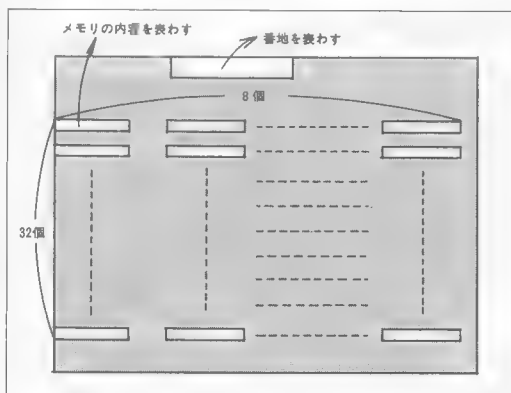
## サブメモリのビット・パターンを見る

■西亀和弘

このプログラムを実行すると図1のようになります。一番上の大きい四角は小さい四角の一番左上の部分の番地の上位2バイト（これ以後は先頭番地とする）を2進コードで表わしています。横8個、縦32個の小さい四角は先頭番地から、+FF<sub>(16)</sub>番地までのメモリの内容を2進コードで表わします。番地の見方は左から右へ+01番地ずつふえていくと考えて、2段めの一番左は先頭番地+08となり一番右下が番地+FF<sub>(16)</sub>となります。

先頭番地の変更としては、先頭番地+100<sub>(16)</sub>は[SPACE]キーをおして[↑]を、先頭番地-100<sub>(16)</sub>は[SPACE]と[↓]、先頭番地+1000<sub>(16)</sub>は[SPACE]キーと[→]、先頭番地-1000<sub>(16)</sub>は[SPACE]キーと[←]です。

図1



ただし、先頭番地が\$D400のときサブCPUが暴走するおそれがあります。先頭番地の初期値は\$D000で\$20CA、20CBに入っています。

使用法として、サブCPUに転送した内容が見たいときは、見たいプログラムのサブ転送の後においてください。またROMの内容を見る場合はこのままEXEC&H2000で走ります。

また、FM-7/8、どちらでも走ります。サブCPU転送される部分は\$20C5-\$2195までで\$7F00以下に転送されます(\$C000に影響を与えないため)。

なお、\$D000はサブCPUのタイマやキー-DATAを見るメモリがあります。

## マシン語ダンプ・リスト

```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
2000 B6 FD 05 2B FA 1A 50 B6 B0 B7 FD 05 8D 5F 7F FD :6F
2010 05 BE 00 00 30 1F 26 FC B6 FD 05 2B FB 33 BD 00 :A2
2020 A4 30 BD 01 79 AF BD 00 9A C6 FF B6 B0 B7 FD 05 :35
2030 B7 FD 05 A6 C0 B7 FC B1 7F FC B0 7F FD 05 7F FD :C3
2040 05 B6 FD 05 2B FB B6 B0 B7 FD 05 B7 FD 05 B6 FC :0D
2050 B0 26 E8 11 A3 BD 00 6B 26 D1 B6 01 B7 FC B0 7F :6A
2060 FD 05 7F FD 05 BE 00 00 30 1F 26 FC 39 10 BE FC :55
2070 B0 30 BD 00 0A C6 43 A6 B0 A7 A0 5A 26 F9 39 00 :6F
2080 00 3F 59 41 4D 41 55 43 4B 49 93 D3 BF 90 B7 D4 :A0
2090 0A B6 D4 0B B6 D4 09 B6 D4 0A CE 7F 00 7F D3 B0 :E2
20A0 7F D3 B0 7F D3 B0 B6 D4 0A B6 D3 B0 27 FB B1 01 :E2
20B0 27 0D B7 D4 0A B7 D4 0A B6 D3 B1 A7 C0 20 DE 7E :4B
20C0 7F 00 01 21 9E 10 CE 3F 00 CE D0 00 BE 00 00 6F :F7
20D0 B9 B0 00 6F B9 40 00 6F B0 BC 3E B0 26 F1 BE 40 :5F
20E0 00 B6 FF A7 80 BC 7E B0 26 F9 10 BE 00 10 1F 30 :52
20F0 34 02 BD 7D 35 02 10 BE 00 B0 BD 75 1F 30 C3 01 :DA

```

Sum: 04 A6 79 35 FD A5 0C 27 D6 E9 32 3F 61 B0 DE 29 :75

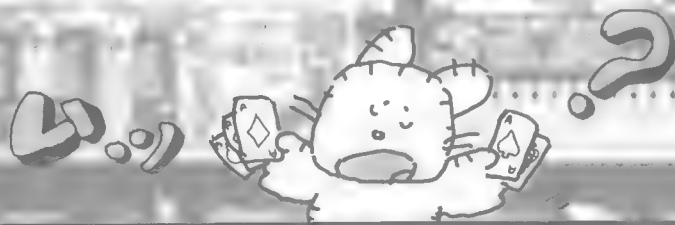
```

Add +0 +1 +2 +3 +4 +5 +6 +7 +8 +9 +A +B +C +D +E +F Sum
2100 00 ED BD 00 91 1F 31 10 BE 02 30 A6 B0 BD 62 31 :71
2110 22 1F 10 C4 07 26 04 31 A9 01 40 AC BD 00 77 26 :37
2120 EA B6 D4 01 B1 20 26 04 6F BD 00 44 E6 BD 00 40 :33
2130 26 D3 B1 E1 27 0E B1 1F 27 15 B1 1C 27 1C B1 1D :27
2140 27 23 20 C1 33 C9 01 00 6C BD 00 24 16 FF 7D 33 :0A
2150 C9 FF 00 6C BD 00 19 16 FF 72 33 C9 10 00 6C BD :66
2160 00 0E 14 FF 67 33 C9 F0 00 6C BD 00 03 16 FF 5C :E3
2170 00 C6 0B 6F A9 40 50 6F A9 40 00 49 34 02 25 0B :7A
2180 6F A9 00 50 6F A0 20 0B 6F A7 A9 00 50 A7 A0 0B :08
2190 35 02 5A 26 DE 39 00 00 00 00 00 00 00 00 00 :CE

```

Sum: C6 36 BA F4 5D B8 2F E1 67 4F 5B 91 77 9D 0E 7B :AB

# VDPカードの製作



森田 昭介

.....  
T I社のビデオ・ディスプレイ・プロセッサ“TMS9918A”(以下VDPと呼ぶ)を使って、FMにスプライト機能を付加するカードを作りました。また、このカードをBASICでコントロールするために、BASICにコマンドを拡張するソフトも制作しました。  
.....

## ハード編

### VDPの特徴

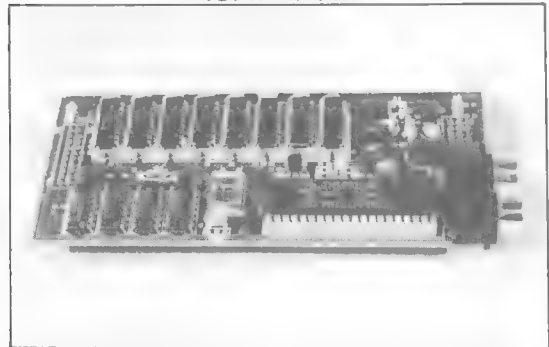
このVDPは他のCRTCと違って、スプライト機能というものを持っています。これはソードのM5、トミーの『ひゅー太』などに使われていて、グラフィックで絵を簡単に動かせるようになっています。

このVDPは大変機能が高く、本体とメモリその他少しの部品で作ることができるのですが、出力はコンホジット出力のみで、RGBのモニタにはつなぐことができません。

### VDPの仕様

- ①256×192ドットのグラフィック
- ②16Kバイトのメモリ・アドレス
- ③16色出力可能(黒白、透明を含む)
- ④重ね合わせ画面の自動処理
- ⑤座標を変更するだけの動画処理

VDPカード本体



#### ⑥1種の静止画モード

このうち、④と⑤については、スプライト画面というものがあり、そこにそれぞれの絵を描いておきます。するとテレビの画面では全部が合成されてみえるわけですが、このとき奥にある(図1)絵は手前の絵と重なったとき、後に隠れてしまいます。この機能を利用すれば三次元的な表現が可能になるわけです。

また、それぞれのスプライトの絵は単に座標を与えてやるだけで、移動や2倍に拡大することもできます。

⑥のモードですが、奥の方にあるキャラクタ・パターン面は4種の表示モードがあります。ふつうこの面は背景などに利用されます。

### スプライト

スプライト面は32面あり#0の面が一番手前にきて、#31が最も奥になります。スプライトのハターンは8×8で256個まで定義でき、そのうち32個を表示できます。

また、1つのハターンを同時に使って16×16のハターンも表示可能で、さらにハターンは2倍の大きさに拡大表示もできます。

色は各スプライト共16色中の1色を表示できます。

### キャラクタ・パターン

キャラクタ・ハターン面には4種類の表示モードを使うことができます。(表1)

#### ■グラフィック・モードI

VDPカード基板装飾

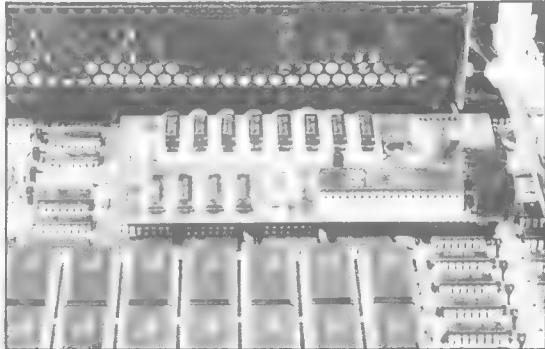


図1 画面構成

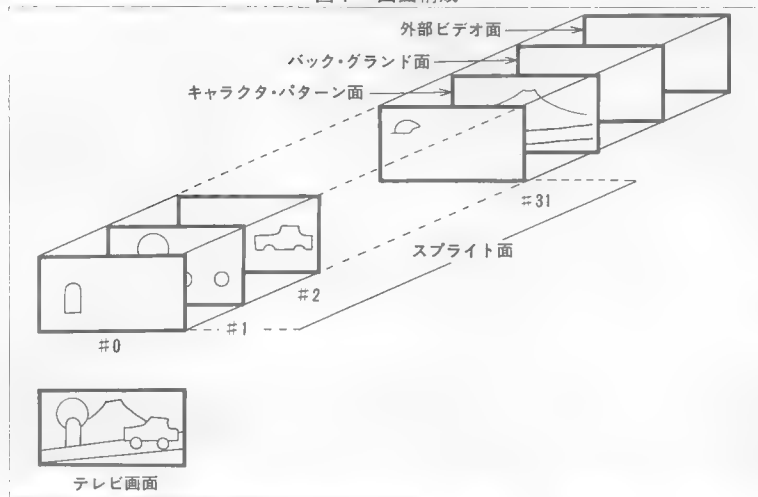


表1 キャラクタ・パターン面のモード

	解像度	パターン・サイズ	パターン	色指定	スプライト	テキスト文字数
グラフィック I	256×192	8×8	256種	16色(8ドットごと)	可	32×24
グラフィック II	256×192	8×8	768種	16色(8ドットごと)	可	32×24
マルチカラー	64×64	4×4(1ブロック)	、	16色(ドットごと)	可	32×24
テキスト	256×192	8×6	256種	16色中2色	不可	40×24

ドット分解能は256×192で256種のパターンを表示可能です。

### ■グラフィック・モード II

ドット分解能は256×192で横8ドットごとに色指定可能です。768種のパターンを表示できます。

### ■マルチカラー

64×48で、1ドットごとに色指定可能です。

### ■テキスト

文字を10×21で表示し、スプライトは使えません。

## 製作の注意

図3の回路図中 CN8A、CN8BはFM-8の場合です。FM-7ではCN9を使います。使用ピン番号は変わりません。もちろんFM-8でエクステンション・ポートを使って作っても良いと思います。

C1～C8はなるべくMB8118(4816コンパチ)のすぐそばに入れます。また、+5V側の足を極力短くしてください。M1～M8の3～7、10～13、15番ピンはハラ(並列)につなぎます。

その他注意する点としてはクリスタルの周辺はTMS9918のすぐそばに配置してください。また、CV1のトリマはしっかりしたものしないと、ちょっとした振動で色ずれを起こします。

製作が終わったらCN8A、CN8Bのそれぞれのピンをテストあたり。ショートしていないことをよく確かめてください。ビデオ出力にCRTをつないで、実験用電源で5Vを供給します。CRTの画面が白っぽくなれば、一応大丈夫でしょう。FM本体に取り付けて、電源を入れます。いままで通り、初期メッセージができればOKでしょう。

図2 パターン定義例

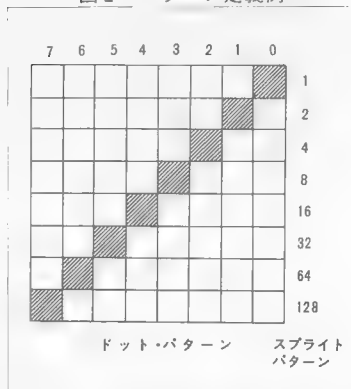


表2 カラーコード表

コード	カラー	コード	カラー
0	透明	8	赤
1	黒	9	淡い赤
2	緑	10	濃い黄
3	淡い緑	11	淡い黄
4	濃い青	12	濃い緑
5	淡い青	13	マゼンタ
6	濃い赤	14	グレー
7	シアン	15	白

## ソフト編

このプログラムはF BASICにリンクして、BASICに新たなVDPコントロール用の命令を拡張するものです。以下にそのコマンドを説明しますが、コマンドは大きく『宣言』、『V-RAM直接アクセス』、『静止画』、『スプライト』の4つに分類されます。

また、マルチフレイクス・スプライトとはソフト的にいくつかのスプライトをまとめて、いっしょに移動させるものです。

なお、FM-7ではリスト1のプログラムの“EXTERNAL DECLARATION”から10行分をリスト2のように変更してからアセンブルしてください。

## 宣言コマンド

宣言コマンドはVDPのモード、スプライトの定義などをするためのコマンドです。

リスト2

9663	FCERR EQU	\$9663
9506	ISDIGC EQU	\$9506
9AD2	EVLAD1 EQU	\$9AD2
9292	CHKCOM EQU	\$9292
98F1	EVLSTR EQU	\$98F1
99BC	GETBY1 EQU	\$99BC
92A0	SNERR EQU	\$92A0
8597	COPBXU EQU	\$8597
C69C	DQSDSP EQU	\$C69C
B50B	FIN EQU	\$B50B







## ■マルチプレックス・スプライトの定義

### 〔機能〕

マルチプレックス・スプライトを定義する。

### 〔フォーマット〕

VDP "MS m=S<sub>n1</sub>, x<sub>1</sub>, y<sub>1</sub>, S<sub>n2</sub>, x<sub>2</sub>, y<sub>2</sub>...S<sub>nk</sub>, x<sub>k</sub>, y<sub>k</sub>"

### 〔パラメータ〕

MS<sub>m</sub>: マルチプレックス・スプライト・ナンバー (m=0~7) の8つまで定義可。

S<sub>nk</sub>: スプライト面番号, n<sub>k</sub>=0~31, VDPのハード的なスプライト面番号を指定する。

x<sub>k</sub>: 水平相対位置

y<sub>k</sub>: 垂直相対位置

## ■キャラクタ・パターンの表示

### 〔機能〕

キャラクタを表示。BASICの"LOCATE+PRINT"と同じ働きをする。

このとき、VDP "GM" で指定したモードにより、1画面の文字数が違う。

### 〔フォーマット〕

VDP "PR<sub>n</sub>=キャラクタ・コード列"

### 〔パラメータ〕

n: グラフィック I、II のとき n=0~767, テキスト・モードのとき n=0~959

キャラクタ・コード列: "CP<sub>n</sub>"のnと同じASCIIコードを持つキャラクタ列。

## ■セット・ドット・カラー指定

### 〔機能〕

セットするドットのカラーを指定する。

### 〔フォーマット〕

VDP "CS=n"

### 〔パラメータ〕

n: カラーコード, n=0~15,

### 〔デフォルト値〕

"CS=15"

## ■リセット・ドット・カラー指定

### 〔機能〕

リセットされるドットのカラーを指定する。

### 〔フォーマット〕

VDP "CR=n"

### 〔パラメータ〕

n: カラーコード, n=0~15

### 〔デフォルト値〕

"CR=0"

## ■テキスト・キャラクタ・カラー指定

### 〔機能〕

テキスト・モードでキャラクタのカラーを指定する (VDP "GM" でテキスト・モードを指定してなければ無効)。

### 〔フォーマット〕

VDP "CT=n"

### 〔パラメータ〕

n: カラーコード, n=0~15

なお、背景色はVDP "CB" で指定した色。

## VRAM直接アクセス・コマンド

VDPの持つVRAMをアクセスするときに使います。

## ■VRAMアドレス指定

### 〔機能〕

VDPのVRAMアクセス・アドレスを指定する。

### 〔フォーマット〕

VDP @n

### 〔パラメータ〕

n: アドレス, 読み出し時は&H0000~&H3FFFF, 書き込み時は&H1000~&H7FFFF.

この後、VDP#, VDP=などで読み書きを行なうごとに1つずつアドレスがインクリメントされます。書き込むときは&H1000を足した値にする。

## ■VRAMへの直接書き込み

### 〔機能〕

VRAMへ1バイト直接に書き込む。書き込み後、アドレス (VDP @nで指定したもの) はインクリメントされる。

### 〔フォーマット〕

VDP #n

### 〔パラメータ〕

n: 1バイト・データ, n=0~255

## ■VRAMへの固定データ書き込み

### 〔機能〕

VRAMへ1バイトの固定データを複数書き込む。

### 〔フォーマット〕

VDP =n, m

### 〔パラメータ〕

n: 書き込みデータ, 1バイト

m: 書き込みデータ転送バイト数

nというデータを指定されたアドレスから, mバイト書き込む。m=0~16384 (&H4000)

## ■VRAMへメイン・メモリより転送

### 〔機能〕

メイン・メモリの指定アドレスからの内容をVRAMへ転送。

### 〔フォーマット〕

VDP <n, m

### 〔パラメータ〕

n: メイン・メモリ・アドレス

m: 転送バイト数, m=0~16384 (&H4000)

## ■VRAMからメイン・メモリへの転送

### 〔機能〕

VRAMの現在のアドレスから指定バイト数をメイン・メモリへ転送。

### 〔フォーマット〕

VDP >n, m

### 〔パラメータ〕

n: メイン・メモリ・アドレス, \$1000~5FFFFの範囲を使用する。

m: 転送バイト数, 0~16384

## 静止画用コマンド

静止画面に対してのみ作用しますが、これらの「静止画用」コマンドはVDP “GM” でグラフィックIIモードになっていなければなりません。

### ■画面のクリア

〔機能〕

静止画面を色、パターンを指定して、それで埋める。

〔フォーマット〕

VDP CLS  $n_1$ ,  $n_2$

〔パラメータ〕

$n_1$  : ドット・パターン、1のビットが画面上でドットとしてセット、0のビットがリセットされる。

$n_2$  : 24ビットはセットされるドットのカラーコードとなり、下位1ビットがリセットのカラーコードとなる。

### ■静止画面に点を打つ

〔機能〕

静止画面上の指定座標にドットをセットする。

〔フォーマット〕

VDP PSET (x, y), C

〔パラメータ〕

x : 水平座標、0 ~ 255

y : 垂直座標、0 ~ 191

C : カラーコード、0 ~ 15

カラーコードは省略可。省略時は前回のときと同じ値となる。ただし、ドットは1キャラクタ内の8ドットが同じ色となる。

〔デフォルト値〕

C = 12

### ■点を消す

〔機能〕

静止画面上の点を指定した色でリセットする。

〔フォーマット〕

VDP PRESET (x, y), C

〔パラメータ〕

ドットをリセットする以外はVDP PSETと同じ。

〔デフォルト値〕

C = 0

### ■線を引く

〔機能〕

静止画面上に線や箱を描く。

〔フォーマット〕

VDP LINE ( $x_1$ ,  $y_1$ ) ( $x_2$ ,  $y_2$ ),  $n_1$ , C,  $n_2$

〔パラメータ〕

$x_1$ ,  $x_2$  : 水平座標、0 ~ 255

$y_1$ ,  $y_2$  : 垂直座標、0 ~ 191

$n_1$  : ドット指定

0	PSET
1	PRESET

C : カラーコード、0 ~ 16 (省略可)

$n_2$  : 線、箱の指定 (省略可)

0	2つの点を通る直線を引く。
1	2つの点を角とする塗りつぶされた箱を描く。

〔デフォルト値〕

省略したパラメータは前回と同じものが適用される。

## スプライト・コマンド

スプライト・コマンドはスプライトのパターン・ナンバー座標を指定するものです。

### ■スプライトの表示

〔機能〕

$n$  のスプライト・フレームにスプライトを表示する。

〔フォーマット〕

SPRITE  $n_1$ , (x, y),  $n_2$ , C

〔パラメータ〕

$n_1$  : 表示するスプライト画番号、0 ~ 31

x : 水平座標、0 ~ 255 (省略可)

y : 垂直座標、0 ~ 191 (省略可)

$n_2$  : パターン・ナンバー (VDP “SP” で定義したもの)  
16×16パターンを指定したとき (VDP “GM”) は0から1つおきのナンバーのみ有効で、その連続した1つのパターンが使われる。それ以外のナンバーはそのナンバーより小さい有効なナンバーとみなされる。

C : 表示カラー、カラーコードに128を足すとスプライトが横に32ドット分シフトする。

### ■マルチプレックス・スプライトの表示

〔機能〕

マルチプレックス・スプライトを表示する。

〔フォーマット〕

SPRITE  $m$ , n, (x, y)

〔パラメータ〕

$n$  : マルチプレックス・スプライト・ナンバー (VDP “MS”) を指定する、 $n = 0 \sim 7$

x : 水平座標、0 ~ 255

y : 垂直座標、0 ~ 191

それぞれのスプライト面に表示されるパターン・ナンバー・カラーはSPRITE  $n$  であらかじめ座標を省略して設定しておく。

リスト1 VDPコントロール・プログラム

		NAM	FJVDP			0077	FACLO	EDU	\$77
		OPT	S			00B4	ARGMD	EDU	\$B4
		TTL	FM-8 SPRITE, VDP						
6000		ORG	\$6000						
		*							
		*	VDP SPRITE COMMAND						
6000 7E	61B7	START JMP	VDPSTR						
	00BC	SKIP2 EDU	\$BC						
		*							
	00D9	TXTPTR EDU	\$D9			9B03	FCERR EDU	\$9B03	
	00D2	CHRGOT EDU	\$D2			99A7	ISDIGC EDU	\$99A7	
	00D8	CHRGOT EDU	\$D8			9E8F	EVLAD1 EDU	\$9E8F	
	01F9	STU81 EDU	\$1F9			9732	CHKCOM EDU	\$9732	
	01FC	STMD1 EDU	\$1FC			9D86	EVLSTR EDU	\$9D86	
		*				9E49	GETBY1 EDU	\$9E49	
		*				9740	SNERR EDU	\$9740	
		*				8A14	COFBXU EDU	\$8A14	
		*				C9D8	DOSDSP EDU	\$C9D8	
	0076	FACMD EDU	\$76			BB80	FIN EDU	\$BB80	



## リスト 1 VDPコントロール・プログラム

[illegible]

## リスト1 VDPコントロール・プログラム

63FB BD 659C JSR GETNUM	64F0 BD 6581 JSR GETC	
63FB F6 6022 LDB MODE	64F1 25 20 6515 BCS ERRFC2	
63FE C1 #1 CMPB #1	64F5 B1 3D CHEA #1	
6400 26 15 6417 BNE VDP706	64F7 26 1C 6515 BNE ERRFC2	
6402 B1 03 CMPA #3	64F9 25 06 PULS #	
6404 26 05 6408 BNE VDP707	64FB BE 6025 LDX PATADR	
6406 BE 0000 LDX #0000	64FE 30 B6 LEAX D,X	
6409 20 07 6412 BRA VDP708	6500 1F 10 TFR X,D	
640B B1 07 VDP707 CMPA #7	6502 1E B9 EXG A,B	
640D 26 4F 645E BNE ERRFC1	6504 CA 40 ORB #540	
640F BE 2000 LDX #2000	6506 BD 624A JSR SREG	
6412 B7 601E VDP708 STA REG4	6509 F6 6003 LDB VTEXTC	
6415 20 0D 6424 BRA VDP709	650C 4F 6003 CLRA	
6417 B7 601E VDP706 STA REG4	650D 1F 01 TFR D,X	
641A BE 0000 LDX #0	650F FE 6004 LDU VTEXTP	
641D 30 B9 0800 VDP7B4 LEAX 204B,X	6512 7E 6266 JMP STMVDP	
6421 4A DECA	6515 7E 9B03 ERRFC2 JMP FCERR	
6422 26 F9 641D RNE VDP7B4		
6424 BF 6029 VDP709 STX P7B4DP		
6427 7E 83A0 JHP VDP7B1		
642A B1 53 VDP7B5 CMPA #5		
642C 26 16 6444 BNE VDP7BR		
642E BD 659C JSR GETNUM		
6431 B7 601F STA REG5		
6434 BE 0000 LDX #0		
6437 30 B9 00B0 VDP7B5 LEAX 12B,X		
643B 4A DECA		
643C 26 F9 6437 BNE VDP7B5		
643E BF 602B STX SPADR		
6441 7E 63A0 JHP VDP7B1		
6444 B1 52 VDP7BR CMPA #2		
6446 26 16 645E BNE ERRFC1		
644B BD 659C JSR GETNUM		
644B B7 6020 STA REG6		
644E BE 0000 LDX #0		
6451 30 B9 0800 VDP7B6 LEAX 204B,X		
6455 4A DECA		
6456 26 F9 6451 BNE VDP7B6		
645B BF 602D STX SPADR		
645B 7E 63A0 JHP VDP7B1		
645E 7E 9B03 ERRFC1 JMP FCERR		
* VDF BACY COLOR SET		
6461 BD 6581 VDFCB1 EQU #		
6464 25 FB 645E JSR GETC		
6466 B1 3D BCS ERRFC1		
646B 26 F4 645E CMPA #1		
646B BD 659C BNE ERRFC1		
646D B4 0F ANDA #50F		
646F F6 6021 LDB REG7		
6472 CA F0 ANDE #5F0		
6474 34 04 VDFCB1 PSHS #		
6476 A4 0E ORA #5+		
647B B7 6021 STA REG7		
647B 7E 6237 JMP INTREG		
* VDF TEXT COLOR SET		
647E BD 6581 VDFCT EQU #		
6481 25 DB 645E JSR GETC		
6485 B1 3D CMPA #1		
6485 26 D7 645E BNE ERRFC1		
6487 BD 659C JSR GETNUM		
648A B4 0F ANDA #50F		
648C 4B ASLA		
648D 4B ASLA		
648E 4B ASLA		
648F 4B ASLA		
6490 F6 6021 LDB REG7		
6493 C4 0F ANDB #50F		
6495 20 DD 6474 BRA VDFCB1		
* VDF SPRITE PATTERN SET		
6497 BD 6497 VDFSP EQU #		
649A 34 02 PSHS A		
649C BD 6581 JSR GETC		
649F 25 74 6515 BCS ERRFC2		
64A1 B1 3D CMPA #1		
64A5 26 70 6515 BNE ERRFC2		
64A5 B6 6003 LDB VTEXTC		
64A8 B1 0B CMPA #50B		
64AA 26 69 6515 BNE ERRFC2		
64AC 35 04 PULS B		
64AE BE 602D LDX SPADR		
64B1 4F VDFSP1 CLRA		
64B2 5B ASLB		
64B3 49 ROLA		
64B4 5B ASLB		
64B5 49 ROLA		
64B6 5B ASLB		
64B7 49 ROLA		
64B8 30 B8 LEAX D,X		
64BA 1F 10 TFR X,D		
64BC 1E B9 EXG A,B		
64BE CA 40 ORB #540		
64CO BD 624A JSR SREG		
64C3 BE 000B LDX #B		
64C6 FE 6004 LDU VTEXTP		
64C9 7E 6266 JMP STMVDP		
* VDF CHR PATTERN SET		
64CC 26 47 6515 BNE ERRFC2		
64CE BD 659C VDFCP EQU #		
64D1 34 02 PSHS A		
64D3 BD 6581 JSR GETC		
64D6 25 7D 6515 BCS ERRFC2		
64D8 B1 3D CMPA #1		
64DA 26 39 6515 BNE ERRFC2		
64DC B6 6003 LDB VTEXTC		
64DF B1 0B CMPA #50B		
64E1 26 32 6515 BNE ERRFC2		
64E3 35 04 PULS B		
64E5 BE 6029 LDX P7B4DP		
64E8 7E 6481 JMP VDFSP1		
* VDF CHARACTER OUTPUT		
64EB BD 64EB VDFPPR EQU #		
64EE BD 65A6 JSR GETNUX		
64EE 34 10 PSHS X		
64F0 BD 6581 JSR GETC		
64F1 25 20 6515 BCS ERRFC2		
64F5 B1 3D CHEA #1		
64F7 26 1C 6515 BNE ERRFC2		
64F9 25 06 PULS #		
64FB BE 6025 LDX PATADR		
64FE 30 B6 LEAX D,X		
6500 1F 10 TFR X,D		
6502 1E B9 EXG A,B		
6504 CA 40 ORB #540		
6506 BD 624A JSR SREG		
6509 F6 6003 LDB VTEXTC		
650C 4F 6003 CLRA		
650D 1F 01 TFR D,X		
650F FE 6004 LDU VTEXTP		
6512 7E 6266 JMP STMVDP		
6515 7E 9B03 ERRFC2 JMP FCERR		
* VDF MULTIPLEX SPRITE SET		
651B BD 651B VDFMSF EQU #		
651C CE 60AF JSR GETNUM		
651E 4D 60AF LDU #MSTAR		
651F 27 06 6527 BEO VDFMS2		
6521 35 2B 1B VDFMS1 DECA		
6524 4A DECA		
6525 26 FA 6521 BNE VDFMS1		
6527 BD 65B1 VDFMS2 JSR		
652A B1 3D CMPA #		
652C 26 E7 6515 BNE ERRFC2		
652E 1F 32 TFR U,Y		
6530 31 23 LEAY 3,Y		
6532 6F C4 CLR ,U		
6534 BD 6534 VDFMS3 EQU #		
6537 B1 53 JSR GETC		
6539 26 DA 6515 BNE ERRFC2		
653B BD 659C BNE GETNUM		
653E A7 A0 STA ,Y+ STORE SPRITE #.		
6540 BD 6581 JSR GETC		
6543 B1 2C CMPA #		
6545 BD 6565 JSR GETNEG		
6548 A7 A0 STA ,Y+ STORE X		
654A BD 6581 JSR GETC		
654D B1 2C CMPA #		
654F 26 C4 6515 BNE ERRFC2		
6551 BD 6565 JSR GETNEG		
6554 A7 A0 STA ,Y+ STORE Y		
6556 6C C4 INC ,U		
6558 7D 6003 TFR VTEXTC		
655B 27 23 6580 BEO GETNG3		
655D A6 C4 LDB ,U		
655F B1 09 CMPA #9		
6561 26 D1 6534 BNE VDFMS3		
6563 20 B0 6515 BRA ERRFC2		
6565 BD 6581 GETNEG JSR		
6568 7F 600B CLR MSIN		
656B B1 2D CMPA #		
656D 26 05 6574 BNE GETNG1		
656F 73 600B CDM MSIN		
6572 20 03 6577 BRA GETNG2		
6574 BD 650B GETNG1 JSR		
6577 BD 659C GETNG2 JSR		
657A 7D 600B TST MSIN		
657D 27 01 6580 BEO GETNG3		
657F 40 NEGA		
6580 39 GETNG3 RTS		
* GET 1 CHARACTER FOR TEXT BUFFER		
* OUT CARRY ON CHARACTER EMPTY.		
* OFF CHARACTER.		
6581 B6 6003 GETC EQU #		
6584 26 03 6589 LDB VTEXTC		
6586 1A 01 6589 BNE GETC1		
6588 39 SEC NO.		
6589 4A 6589 GETC1 EQU #		
658A B7 6003 DECA		
658B BE 6004 STA VTEXTP		
658C A6 B4 LDX VTEXTP		
658E 30 01 LEAX 1,X		
6594 BF 6004 STX VTEXTP		
6597 B1 20 CMPA #520		
6599 27 E6 6581 BEO SPACE.		
659B 39 BEQ SKIP IT.		
* GET NUMBER		
659C BD 659C GETNUM EQU #		
659F 0D 76 JSR GETNUX		
65A1 26 50 65F3 TST FACMO		
65A3 96 77 BNE ERR3		
65A5 39 LDB FACLO		
65A6 BD 65A6 GETNUX EQU #		
65A9 BD 6581 JSR SETXTP		
65AC 25 07 6585 GETNU1 JSR		
65AE BD 99A7 BCS GETNU2		
65B1 24 F6 65A9 JSR ISDIG		
65B3 BD 26 650B BCC GETNU1		
65B5 BE 6004 BSR ENCHNL		
65B8 A6 B4 GETNU2 LDB		
65BA 34 52 LDB ,X		
65BC 6F 84 CLR A,X,U		
65BE BD B8B0 JSR FIN		
65C1 35 32 PULS A,U,X		
65C3 A7 B4 STA ,X		
65C5 BD 65EA JSR RECP		
65C8 9E 76 LDX FACMO		
65CA 39 RTS		
65CB 34 10 SETXTP PSHS X		
65CD 9E D9 LDX X		
65CF BF 6006 STX VTEXTP		
65D2 BE 6004 LDX VTEXTP		
65D5 3D 1F LEAX -1,X		
65D7 9F D9 STX XTPTR		
65D9 35 90 PULS X,PC		
65DB 34 10 ECHNL PSHS X		
65DD 7C 6003 INC VTEXTC		



## リスト 1 VDPコントロール・プログラム

[illegible]

67DD B4	OF	AND	#50F	689F CA	40	ORR	#540
67DE AB	E0	ORA	.S+	68A1 BA	524A	JSR	SREG
67E1 34	02	PSHS	A	68A4 35	02	PULS	A
67E3 FC	6027	DD	COLADR	68A6 BE	1800	LDR	#2048#3
67E6 33	8B	LEAU	D, X	68A9 BD	6251	JSR	STCVDP
67EB 1F	30	TFR	U, D	68AC BD	9732	JSR	CHKCOM
67EA 11	89	EXG	A, B	68AF BD	5221	JSR	GETBYT
67EC CA	40	ORR	#540	68B2 34	04	PSHS	B
67EE BD	624A	JSR	SREG	68B4 FC	6027	LDD	COLADR
67F1 35	02	PULS	A	68B7 1E	89	EXG	A, B
67F3 B7	FDCE	STA	VDATA	68B9 CA	40	ORR	#540
67F6 39		RTS		68BB BD	624A	JSR	SREG
				68BE 3E	02	PULS	A
				68C0 BE	1800	LDR	#2048#3
67F7 BD	40	CALBAS	EDU	68C3 BD	6251	JSR	STCVDP
67F9 1F	10	6846	CALNAM	68C6 FC	6025	LDD	PATADR
67FB 1E	89	TFR	X, D	68C9 1E	89	EXG	A, B
67FD BD	624A	JSR	SREG	68CB CA	40	ORR	#540
6800 F6	FDCE	LDB	VDATA	68CD BD	624A	JSR	SREG
6803 34	04	PSHS	B	68D0 C6	03	LDB	#3
				68D2 4F		CLRA	
6805 BE	0000	LDR	#0	68D3 B7	FD2E	VDPL1	VDATA
6808 F6	600A	LDB	PY	68D6 AC		INCA	
680B 54		LSRE		68D7 26	FA	68D3	BNE
680C 54		LSRE		68D9 5A		DEC8	
680D 54		LSRE		68DA 26	F7	68D3	BNE
680E 54		LSRE		68DC 9D	DB		CHRGOT
680F 54		LSRE		68DE 26	82	6892	BNE
6810 54		LSRE		68E0 39			ERRFC9
6811 27	07	681A	BE0				
6813 30	89	0800	CALBA1				
6815 54		LEAX	2048, X				
6818, 26	F9	6813	DEC8				
	681A	CALBA2	BNE				
681A 35	04	PULS	B				
681C 4F		CLRA					
681D 58		ASLE					
681E 49		ROLA					
681F 58		ASLE					
6820 49		ROLA					
6821 58		ASLE					
6822 49		ROLA					
6823 30	8B	LEAX	D, X				
6825 F6	600A	LDB	PY				
6828 BD	10	683A	MOD8				
682A 3A		ARX					
682B 39		RTS					
682C F6	6009	BITSHT	LDB				
682F 86	80	07	683A				
6831 BD	07	683A	BSR				
6833 27	04	6839	BE0				
6835 44		BITSHT	LSRA				
6836 5A		DEC8					
6837 26	FC	6835	BNE				
6839 39		BITSHT	RTS				
683A 34	04	683A	MOD8				
683C 54		PSHS	B				
683D 54		LSRE					
683E 54		LSRE					
683F 58		ASLE					
6840 58		ASLE					
6841 58		ASLE					
6842 50		NEGE					
6843 EB	E0	ADD	.S+				
6845 39		RTS					
6846 BE	6846	CALNAM	EDU				
6849 F6	600A	LDR	PY				
684C 54		LSRE					
684D 54		LSRE					
684E 54		LSRE					
684F 86	20	LDA	#32				
6851 3D		MUL					
6852 30	8B	LEAX	D, X				
6854 F6	6009	LDB	PX				
6857 54		LSRE					
6858 54		LSRE					
6859 54		LSRE					
685A 3A		ARX					
685B 39		RTS					

## リスト 1 VDPコントロール・プログラム

6A68 7F	6011	CLR	SY			6A62 B6	600C	LDA	X1		
6A6E 2A	06	PSHS	D			6A65 B7	600E	STA	X2		
6A90 FA	600F	LDB	Y2			6A68 F7	600C	STB	X1		
6A93 A3	E1	SUBD	.S++			6A6A F7	6009	STB	PX		
6A95 27	08	69C2	BEQ	LINE4		6A6E BD	6B46	HSLIN1	JSR	CALNAM	
6A97 2A	06	69BF	BPL	LINE3		6A71 3A	10	X	PSHS		
6A9D 50		NEGB				6A73 F6	600E	LDB	X2		
6A9A 7A	6011	DEC	SY			6A76 F7	6009	STB	PX		
6A9D 20	03	69C2	BRA	LINE4		6A79 BD	6B46	JSR	CALNAM		
6A9F 7C	6011	INC	SY			6A7C 1F	10	TFR	X,D		
69C2 F7	6012	LINE4	STB	DY	DY=Y2-Y1	6A7E A3	E1	SUBD	.S++		
6A9C FC	600C	LDD				6A80 5A		DECB			
6A9B FD	6009	STD	PX			6A81 F7	6012	STB	DX		
6A9B 7D	6010	TST	SX			6A84 F6	600C	LDB	X1		
69C2 27	3F	6A0F	BEQ	VLIN		6A87 F7	6009	STB	PX		
6A9D 70	6011	TST	SY			6A8A BD	67F7	JSR	CALBAS		
6A9D 37	4E	6A20	BEQ	HLIN		6A8D BD	4E	6ADD	BSR	PXMDOB	
6A9D CC	0000	LDD	#0			6A8F 26	03	6A94	BNE	HSLIN2	
6A9D BD	6016	LTD	LY			6A91 4F		CLRA			
6A9D BD	70	6A4D	BSR	DOLINE		6A92 20	08	6A9C	BRA		
6A9D F6	6013	LDB	DY			6A94 B6	80	HSLIN2	LDA	##90	
6A9D 4F		CLRA				6A96 20	01	6A99	BRA	HSLIO3	
6A9E1 F3	6016	ADD	LY			6A98 4A		HSLIN3	LSRA		
6A9E4 FD	6016	STD	LY	LY=LY+DY		6A99 5A		HSLIO3	DECB		
6A9E7 F6	6012	LINE5	LDB	DX	Deq=DX	6A9A 26	FC	6A9B	HSLIN3		
6A9E8 4F		CLRA				6A9C 4A		HSLIN4	DECA		
6A9E8 3A	06	CLRA	PSHS	D		6A9D BD	33	6AD2	BSR	HSLIN5	
6A9ED FC	6016	LDD	LY			6A9F 7D	6012	TST	DX		
6A9F0 A3	E1	SUBD	.S++	Deq=LY-DX		6AA2 27	16	6ABA	BEQ	HSLIN6	
6A9F2 25	09	69FD	BDS	LINE52		6AA4 F6	6009	HSLIN5	LDB	PX	
6A9F4 FD	6016	STD	LY			6AA7 4F		CLRA			
6A9F7 BD	47	6A40	BSR	INCPY		6AAB C3	000B	ADD	##0		
6A9F9 BD	52	6A4D	BSR	DOLINE		6AAB F7	6009	STB	PX		
6A9FB 20	EA	6A9E7	BRA	LINE51		6AAE BD	67F7	JSR	CALBAS		
6A9FD 26	08	6A07	LINE52	BNE	LINE53	6AB1 B6	FF	LDA	##FF		
6A9FF CC	0000	LDD	#0			6AB3 BD	1D	6AD2	BSR	HSLIN5	
6A02 FD	6016	STD	LY			6AB5 7A	6012	DEC	DX		
6A05 BD	39	6A40	BSR	INCPY		6AB8 26	EA	6AA4	BNE	HSLIN5	
6A07 BD	2A	6A33	LINE53	BSR	INCPY	6ABA F6	600E	HSLIN6	LDB	X2	
6A09 7A	6015	DEC	LP			6ABD F7	6009	STB	PX		
6A0C 26	CD	6A9B	BNE	LINE50		6ACO BD	67F7	JSR	CALBAS		
6A0E 39		RTS				6AC3 B6	80	LDA	##80		
						6AC5 BD	16	6ADD	BSR	PXMDOB	
						6AC7 27	06	6ACF	BEQ	HSLIN9	
						6AC9 1A	01	HSLIN7	SEC		
						6ACB 46		RORA			
						6ACC 5A		HSLIN8	DECB		
						6ACD 26	FA	6AC9	BNE	HSLIN7	
						6ACF BD	01	6AD2	HSLIN9	BSR	HSLIN5
						6AD1 39		RTS			
						6AD2 7D	6014	HSLIN5	TST	PRSF8	
						6AD5 26	03	6ADA	BNE	HSLINR	
						6AD7 7E	6722	JMP	VDPFS2		
						6ADA 7E	679E	HSLINR	EQU		
								JMP	VDPFR2		
						6ADD F6	6009	PXMDOB	LDB	PX	
						6AE0 7E	6B3A	JMP	MODB		

## リスト 3 デモ・プログラム

```

5 CLEAR,&H$FFFF:LOADM"FJVDP",,R
10 A=3.14159/16:A1=A/3:A2=A*2:A3=A*1.5
20 VDP"CB=14"
30 VDP"GM=M1S3":VDPCLS0,&H70:FORI=50TO160:VDPLINE(160-I,I+10)-(94+I,I+10),0,3:NE
XT
40 VDPLINE(111,61)-(143,61),1,15
50 VDPLINE(143,61)-(250,168),1,14
60 VDPLINE(128,61)-(128,168),1,14
70 VDPLINE(111,61)-(5,168),1,14
80 VDPLINE(5,168)-(250,168),1,14
90 VDPLINE(160-84,10+86)-(94+86,10+86),0,15
100 VDPLINE(160-84,86)-(94+86,86),0,15
110 VDPLINE(76,86)-(76,96),0,15
120 VDPLINE(180,86)-(180,96),0,15
140 VDP"SP8="+CHR$(&HOF)+CHR$(&H3F)+CHR$(&H7F)+CHR$(&H7F)+CHR$(&HFF)+CHR$(&HFF)+
CHR$(&HFF)+CHR$(&HFF)
150 VDP"SP9"+"&H$(&HFF)+CHR$(&HFF)+CHR$(&HFF)+CHR$(&HFF)+CHR$(&HFF)+CHR$(&HFF)+CHR$(&HEE)+"

```

```

CHR$(%HC6)+CHR$(%H82)
160 VDP"SP10="+CHR$(%HF0)+CHR$(%HFC)+CHR$(%HFE)+CHR$(%HFE)+CHR$(%HFF)+CHR$(%HFF)+
CHR$(%HFF)+CHR$(%HFF)
170 VDP"SP11="+CHR$(%HFF)+CHR$(%HFF)+CHR$(%HFF)+CHR$(%HFF)+CHR$(%HFF)+CHR$(%H77)+
CHR$(%H63)+CHR$(%H41)
180 VDP"SP4="+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H31)+CHR$(%H7B)+
CHR$(%H7B)+CHR$(%H31)
190 VDP"SP5="+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+
CHR$(%H00)+CHR$(%H00)
200 VDP"SP6="+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H80)+CHR$(%HC0)+
CHR$(%H80)+CHR$(%H80)
210 VDP"SP7="+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+
CHR$(%H00)+CHR$(%H00)
220 VDP"SP0="+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H63)+
CHR$(%H63)+CHR$(%H00)
230 VDP"SP1="+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+
CHR$(%H00)+CHR$(%H00)
240 VDP"SP2="+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+
CHR$(%H00)+CHR$(%H00)
250 VDP"SP3="+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+CHR$(%H00)+
CHR$(%H00)+CHR$(%H00)
260 SPRITE2,(0,0),8,4
270 SPRITE1,(100,0),4,15
280 SPRITE0,(100,0),0,1
290 SPRITE5,(0,0),8,8
300 SPRITE4,(100,0),4,15
310 SPRITE3,(100,0),0,1
320 SPRITE8,(0,0),8,10
330 SPRITE7,(100,0),4,15
340 SPRITE6,(100,0),0,1
350 SPRITE9,(100,%H80),0,1
360 VDP"MS0=S0,0,0S1,0,0S2,0,0"
370 VDP"MS1=S3,0,0S4,0,0S5,0,0"
380 VDP"MS2=S6,0,0S7,0,0S8,0,0"
390 FOR X=0TO255
400 X1=SIN(A1*X)*100
410 X3=SIN(A2*X)*100
420 X4=SIN(A3*X)*100
430 X2=COS(A*X+B)*70
440 IF X MOD 16 >8 THEN VDP"MS0=S0,0,0 S1,0,0 S2,0,0" ELSE VDP"MS0=S0,0,7S1,0,5S
2,0,0"
450 IF X MOD 16 >8 THEN VDP"MS1=S3,0,0 S4,0,0 S5,0,0" ELSE VDP"MS1=S3,0,7S4,0,5S
5,0,0"
460 IF X MOD 32 >16 THEN VDP"MS2=S6,0,0 S7,0,0 S8,0,0" ELSE VDP"MS2=S6,0,7S7,0,5
S8,0,0"
470 IF X MOD 64 >32 THEN VDP"GM=S2" ELSE VDP"GM=S3"
480 SPRITE00,(X4+110,80+X2/10)
490 SPRITE01,(X2+110,140+X1/5)
500 SPRITE02,(X3+110,20+X1/5)
510 NEXT:B=RND*10:GOTO390

```

## R A N D O M B O X

### 2dot SCROLL

#### ■JG3ECZ

FM-8の両面スクロールはオフセットアドレスを変更することによってメモリ転送せずにハード的に行なっています。これをうまく使えば、計算上2ドットのスクロールが可能になります。

USR関数の整数型引数により、次の機能に分かれます。

- 0:上シフト
- 1:上ローテイト
- 2:下シフト
- 3:下ローテイト
- 他:変化なし

ただし、実行後カーソルの移動などにより、キャラクタ用バッファ・メモリの文字が再び表われます。100回繰り返すことで、画面クリア、または一周します。

なお、プログラムはポジション・インディペンデントです。

\$6F1Bからの2バイトをF1, 7DにすればFM-7でも動きます。

#### 2dot SCROLL

Add	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	Sum
6F00	34	37	EC	02	10	83	00	04	24	13	E7	8D	00	24	30	8D	:7E
6F10	00	13	AF	8D	00	0B	30	8D	00	05	BD	F2	0B	35	B7	10	:9F
6F20	00	00	00	00	00	77	00	00	3F	59	41	4D	41	53	43	48	:07
6F30	93	D3	90	90	00	34	77	7D	D4	09	B6	50	8E	00	00	10	:FF
6F40	BE	3E	80	E6	BC	EE	C4	02	27	07	BE	3D	E9	10	BE	3F	:28
6F50	60	E6	BC	E0	C4	01	27	19	EE	89	80	00	EF	A9	80	00	:C6
6F60	EE	89	40	00	EF	A9	40	00	EE	B1	EF	A1	4A	26	E9	20	:07
6F70	10	CE	00	00	EF	A9	80	00	EF	A9	40	00	EF	A1	4A	26	:CE
6F80	F3	FC	D0	1D	C3	00	A0	66	BC	AA	66	BC	A7	24	03	B3	:1E
6F90	01	40	B4	3F	FD	D0	1D	FD	D4	0E	35	F7	00	00	00	00	:F9
-----																	
Sum:	A7	D4	CB	41	75	D3	0F	CB	A3	D4	4F	71	6A	42	73	FE	:FD

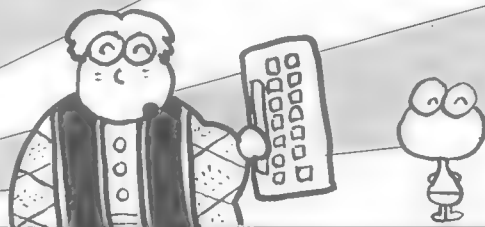
#### 上ローテイト・サンプル

```

10 DEF USR0=%H6F00
20 FOR I=0 TO 1000
30 A=USR(1)
40 NEXT

```

# リアルタイム キースキャン



■長川 茂

サブシステムの使い方、BIOSの使い方がわかるにつれ、高速ゲームもいくつか発表されました。しかし、FM-8のキー入力が一番最後に押されたキーの内容がいつまでも残り、いまのところソフトでキャンセルすることはできません。このことはゲームでは致命傷となり、手をはなしてもビーム砲などがかってに移動してしまいます。そこで、誰でも簡単にできるわずかな改造でリアルタイムのキースキャンを可能にしたので発表します。

## キー入力について

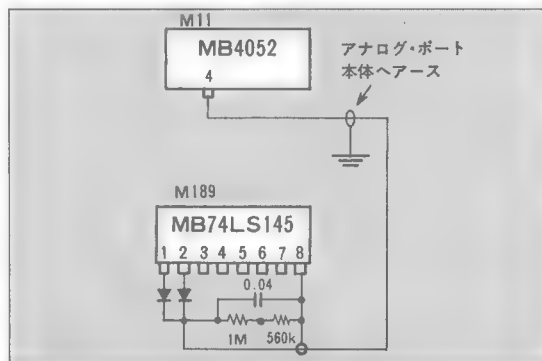
FM-8のキーはまずMB8841という4ビットのキーボード専用マイコンに読みとられ、PFキーを含むほとんどのキーをASCIIコードに直して出力側に送ります。つぎにこのマイコンがパルスを送り、サブCPUにFIRQをかけ、入力があったことを知らせます。そしてサブ・モニタが\$D401（アドレスはサブCPU側）をREADするとトライ・ステート・バッファがONになりデータ・バスに乗ります。つまり、トライ・ステートがOFFの状態でもその直前まではデータが出力されたままになっており、これが誤動作の原因になっていました。

リピートは、FIRQをかけるパルスの長さを操作すればよく、初めの入力があったから約0.4秒おいてなお同じキーが入力され続けていれば、その後断続的にパルスを送り出しています。だからこのパルスを操作しないと、最初の入力があったからワンステップおいてしまうためゲームではとても実用にはなりません。

このパルスの間隔はMB8841にプログラムされているので、書き変えることはまず不可能です。そこでこのパルスを操作する回路をTTLで組みましたが、回路が複雑になってしまい、パターン・カットも必要なので、2号機、3号機を作っていくうちあきらめようかとも思いました。そんなある日、ついに非常に簡単な方法を見つけました。アナログ・ポートが大きな口をあけて寝ているので、これを使わない手はないとさっそく図1のような配線をしたところ結果は良好でした。

これでポートの内容をBIOSで読むことによってフラグ代りに使え、ついにリアルタイムの夢が実現しました。

図1



部 品 表

抵抗	1MΩ	1
	560kΩ	1
コンデンサ（マイラ）	0.04μF	1
シールド線	40cmぐらい	1本
ダイオード	1S1588など	2本

## 製 作

これはいたって簡単で、一度でもハンダごてをにぎった人なら5分ぐらいで完成するでしょう。

注意としては配線は必ずシールド線を使ってください。もちろん1m100円ぐらいの切り売り特価品で結構です。電気街へ行けばすぐみつかるでしょう。あとはいつもの抵抗、コンデンサ、ダイオードだけです。

## 調 整

相手がアナログだけに少々調整を必要とします。まず、リスト1を打ち込みRUNさせてください。キーを押しているときは255、離しているときは0に近い数字になっていることを確認します。つぎにキーをすばやくチェンジさ

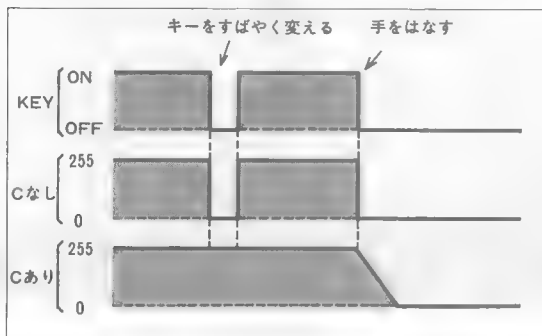
## リスト 1

```

10 A=ANPORT (0, 1)
20 PRINT A;
30 GOTO10

```

図 2



せたとき (たとえばインベーダーゲームのように右へいったらすぐ左へ戻るようなキー操作), チェンジさせた瞬間も255になっていること, キーを離したらすぐに255以下の数字が出ることを確認します。もしこのようにならない場合はCの値を次のように少し変えてみてください。

①すばやくチェンジさせたときに255が変化してしまうとき……増加

②手を離してもしばらく255が続くとき……減少

②の調整を最優先させてください。①の方は多少なら変化してもかまいません。またコンデンサを小さくしすぎた場合, キーをずっと押しているにもかかわらず255が変化してしまうことがあるので注意してください。

なぜこのようなことをするかというと, 素早くチェンジさせたときあまりシビアに反応すると, インベーダーゲームのような左右の切り換えが激しいものでは少々ギクシャクしてしまうからです。そこで急激な変化に対しては入力が続くようにします(図2)。

逆に手を離して入力を継続させない場合はすぐに値を変化させ, 入力が終わったことを知らせなければならないので立ち下りを早くします。以上のことはソフトでも解決できますが(PC-8001などはそうしている), 私の場合はコンデンサを付けることによって実現しています。

## 応 用

今回はテン・キーのみをリアルタイム・スキャン可能にしました。ICの足とキーの種類およびチャネルの関係は表1のようになっているので, それぞれの足から別々のチャネルに接続すれば, 読み出すチャネルによって押されているキーが調べられます。2人や3人でも同時に遊ぶことができます。どうです, こんなに簡単に, しかもキーを別々に判断できるなんてすごいでしょ。もちろん普段の使用にはまったく影響ありません。

これからもっとFM-8のゲームが増えることを希望します。皆さん, FM-8を最高のゲーム・マシンにしようではありませんか!

## 参考文献

- 1) FM-8活用研究
- 2) I/O '82年7月号

表 1 LS145のピンと対応するキー

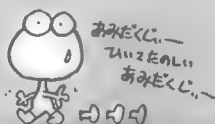
ピン	関係するキー
1	0~7 (テン・キー)
2	8~とRET.X2 (テン・キー)
3	A~G, @
4	H~O
5	P~W
6	X~Z, -, /, *, [ , \
7	0~7
9	8~9, 本, ル, メ, ; , :
10	ESC, TAB, INS, DEL, EL, CLS, HOME, SPACE
11	PF1~PF8

その他はハードの都合上できません

MB4052

ピン	関係するキー
4	0
5	1
6	2
7	3

# JOYSTICKの製作



■DOSのTom

FM-7でゲームをやる場合、キーが押されていない状態を判別できずに苦労している方も多いと思います。そこで、シンプルな接点式のJOYSTICKを製作し良好な結果を得たので、ここに発表させていただきます

## 回路について

回路はいたって簡単で、図1のようなもので充分です。多少、実物と異なっていますが、これは実物の方が4つまでJOYSTICKを接続できるようにしたためです。また、JOYSTICKの各接点とビットの対応が不自然に思われるかもしれませんが、ソフトの負担を軽減するためですからこの通りに接続してください。

## 製作について

JOYSTICKはジャンク屋などで購入します。私は8方向タイプのものを使いましたが、通常のゲームはほとんどが4方向で充分ですから8方向型を使用する必要はないと思います。それに、4方向型の方がソフトも楽ですし、動作も確実なようです。

つぎにスイッチですが、一番力のかかる部分なので頑丈なものを使いましょう。TTL類は一般的なものばかりですが、コネクタは多少入手困難かもしれません。

部品が揃ったら、あとはほとんど板金加工です。最低でもドリル、リーマ、ヤスリなどが必要です。見栄えをよくするためにはJOYSTICKはネジ止めせず、エポキシ系接着剤などで固定するのがよいようです。

## ソフトについて

8方向JOYSTICKで4方向データを読むプログラムをリスト1に示します。キーボードと同様に前のデータを保持するようにしたい場合は、BRA SETKEYをNOPなどに変わってください。

製作例

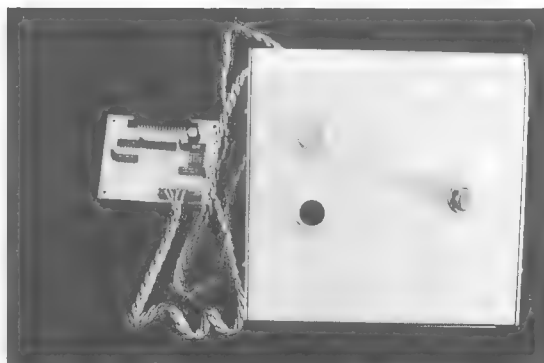
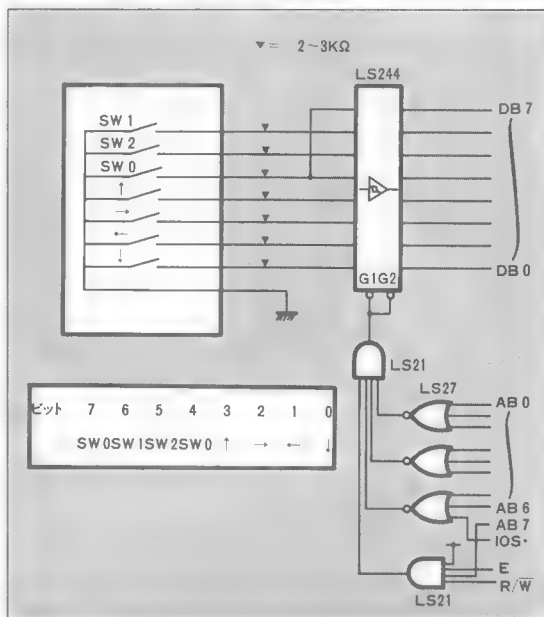


図1 全回路図



## 終わりに



FM-7について一部誤った情報が流れているようなので、気が付いた点を書いておきます。

① ANPORTがなくなった。

BIOS、BASIC共にサポートしており、FM 8と同じADC (産業用のため入手困難)をSF00Cポートに接続すれば利用できます。

② キーの先き読みができない。

BASICでPRINT CHR\$(27)+"g"を実行することにより、32文字の先き読みができる。これはマニュアルにも書いてあるのですが、以外と知らない人が多いようです。また、

FM 7には各種のキーボード制御機能が付いており、先き読みの中断なども可能です (FM 7ユーザーズ・マニュアル・システム仕様書2.31参照)。ここで注意しなければならないことは、“エスケープ・シーケンスは同一フィールド内で定義されなければならない”ことです。ですから、PRINT CHR\$(27)+"g"とやったのでは先き読みモードにはならず、“g”が表示されるのみです。ですから、PRINT CHR\$(27)+"g"とします。

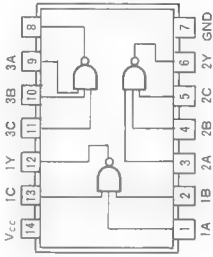
リスト1 JOYSTICK読み込みプログラム

```

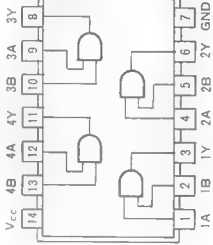
0001 0000      ;
0002 0000      ; FM-7 JOYSTICK CONVERT PROGRAM (Non Repeat)
0003 0000      ;
0004 0000      ; For 4 DST. Then Key (2,4,6,8,5,CR,STOP)
0005 0000      ;
0006 0000      JYPORT=$FD80
0007 0000      ;
0008 2000      ; ORG $2000
0009 2000      ;
0010 2000 3404  CONVRT:PSHS B
0011 2002 8680      LDA #$80          ;Set STOP Key Condition
0012 2004 F6FD80     LDB JYPORT
0013 2007 2A2F      BPL RET          ;bit 7 Head Swich Pressed
0014 2009          ;
0015 2009 860D      LDA #$0D          ;CR Code
0016 200B C540      BITB #$40          ;Check Blue Switch
0017 200D 2729      BEQ RET
0018 200F 8635      LDA #$35          ;LDA #'5'
0019 2011 C520      BITB #$20          ;Check Black Switch
0020 2013 2723      BEQ RET
0021 2015          ;
0022 2015 C1        FCB $C1          ;CMPB #xx OP Code
0023 2016 FF        OLDBIT:FCB $FF
0024 2017 2717      BEQ RECOVR        ;Condition Not Changed
0025 2019 E78DFFF9  STB OLDBIT,PCR
0026 201D          ;
0027 201D 8632      LDA #'2'
0028 201F 54        LOOP:LSRB          ;Key Condition -> Carry
0029 2020 2506      BCS NEXT
0030 2022 A18D000B  CMPA OLDKEY,PCR
0031 2026 260C      BNE SETKEY        ;New Status
0032 2028 4C        NEXT:INCA
0033 2029 4C        INCA
0034 202A 813A      CMPA #$3A
0035 202C 26F1      BNE LOOP
0036 202E          ;
0037 202E          ; If you want to unperssed status
0038 202E          ; then use following statement
0039 202E          ;else don't used.
0040 202E          ;
0041 202E 2004      BRA SETKEY
0042 2030          ;
0043 2030 86        RECOVR:FCB $86    ;LDA #xx OP Code
0044 2031 00        OLDKEY:FCB 0
0045 2032 3584      PULS B,PC
0046 2034          ;
0047 2034 A78DFFF9  SETKEY:STA OLDKEY,PC
0048 2038 3584      RET: PULS B,PC
0049 203A          ;

```

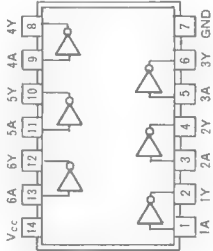
LS10



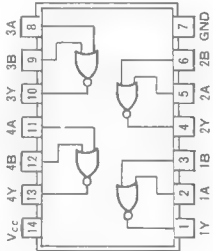
LS08



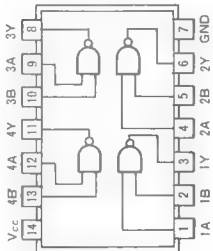
LS04. 05



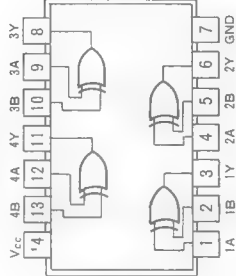
LS02



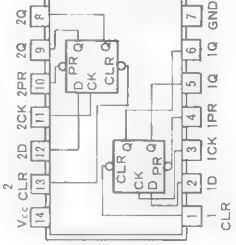
LS00



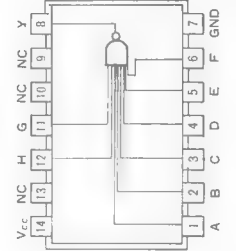
LS86, LS136



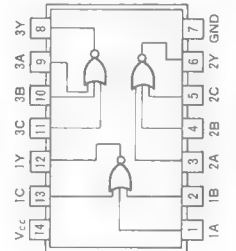
LS74



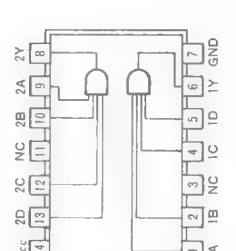
LS30



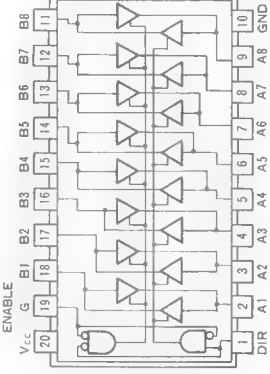
LS27



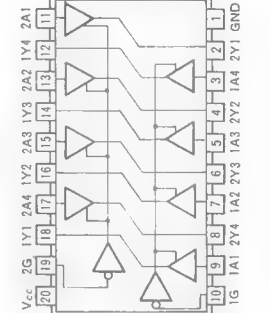
LS21



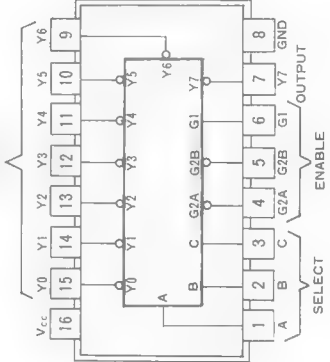
LS245



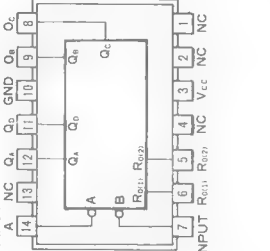
LS244



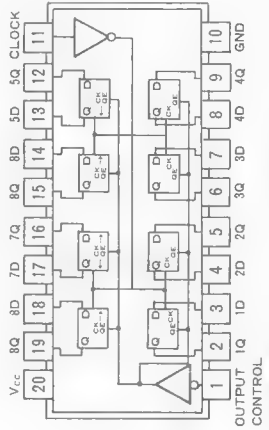
LS138



LS93



LS374



2SC725



- 1 : エミッタ (E)
  - 2 : コレクタ (C)
  - 3 : ベース (B)
- (底面)

## 資料編

FM7サブシステム・モード・ソース・リスト	242
FM-7/BIOSソース・リスト	276
FM-7 全回路図	289
FM-8サブリ・モードとDOSモード	308
FM-8活用研究正誤表	324



```

000001 OPT CAT,MEA
000002 OPT LIST,LLEN=120,CREF
000003 TTL DISPLAY SUB=SYSTEM2
000004 NAM DSUB=JP
000005
000006 DISPLAY SUB=SYSTEM MONITOR
000007
000008 FUNCTION BLOCK
000009
000010 CONSOLE
000011 GRAPHIC
000012 I EYED-AD
000013 TIMEF
000014
000015 FACILITIES
000016 DISPLAY (VRAM 16K*32)
000017 KEYBOARD
000018 20 MS TIMER
000019 BUZZER
000020
000021 DESIGNED BY M.YAMAUCHI
000022
000023 AUG. 9 1982 SUB=SYSTEM2 V6.0 JF. VERSION
000024
000025 CODED BY M.YAMAUCHI
000026
000027 ALL RIGHT REASEVED
000028
000029
000030 SORG MACR
000031 .SORG SET #0
000032 ENDM
000033 DS MACR
000034 #0 EQU .SORG
000035 .SORG SET .SORG+#1
000036 ENDM
000037 FLGENT MACR
000038 #1 FCB $36 LDA #
000039 #0 CLRA
000040 STA #C
000041 ENDM

```

00122			#	I/O	REGISTER
00123			#		
00124A 0000					S0FG
	D400	A	.\$0FG		SET
					DS
00125A 0000		D400	A	IC001	EQU
	D401	A	.\$0FG		SET
					DS
00126A 0000		D401	A	IC002	EQU
	D402	A	.\$0FG		SET
					DS
00127A 0000		D402	A	AC11FQ	EQU
	D403	A	.\$0FG		SET
					DS
00128A 0000		D403	A	BUZZEP	EQU
	D404	A	.\$0FG		SET
					DS
00129A 0000		D404	A	ATTENT	EQU
	D405	A	.\$0FG		SET
					DS
00130A 0000		D405	A	IORDM	EQU
	D408	A	.\$0FG		SET
					DS
00131A 0000		D408	A	CRTON	EQU
	D409	A	.\$0FG		SET
					DS
00132A 0000		D409	A	VACC	EQU
	D40A	A	.\$0FG		SET
					DS
00133A 0000		D40A	A	BUSY	EQU
	D40B	A	.\$0FG		SET
					DS
00134A 0000		D40B	A	IORDM2	EQU
	D40C	A	.\$0FG		SET
					DS
00135A 0000		D40C	A	INSLED	EQU
	D40E	A	.\$0FG		SET
					DS
00136A 0000		D40E	A	S0FFSET	EQU
	D410	A	.\$0FG		SET

```

00041      SI IP0 MACP
00042      FCB FSC CHPX #
00043      ENDM
00044      SI IP1 MACP
00045      FCB $C1
00046      ENDM
00047      PTPUL MACP
00048      PULS #0,PC
00049      ENDM
00050      ACI MACP
00051      BITA ACI-IPQ
00052      ENDM
00053      SCRON MACP
00054      BITA CRTON
00055      ENDM
00056      SCROFF MACP
00057      STA CRTON
00058      ENDM
00059      VACON MACP
00060      BITA VACC
00061      ENDM
00062      VACOFF MACP
00063      STA VACC
00064      ENDM
00065      BSYON MACP
00066      STA BUSY
00067      ENDM
00068      BSYOFF MACP
00069      TST BUSY
00070      ENDM
00071      BUZZON MACP
00072      BITA BUZZEP
00073      ENDM
00074      SYSREQ MACP
00075      BITA ATTENT
00076      ENDM
00077      INSON MACP
00078      BITA INSLD
00079      ENDM
00080      INSOFF MACP

```

00137			*
00138			*
00139		A	AMCL EQU
00140		A	AMRV EQU
00141		0010	A AMPT EQU
00142		0040	A AMMD EQU
00143		0080	A AMFL EQU
00144			*
00145		0050	A CSCX1 EQU
00146		0028	A CSCX2 EQU
00147		0019	A CSCY1 EQU
00148		0014	A CSCY2 EQU
00149		0050	A GBXS2 EQU
00150		0280	A GXD EQU
00151		00C8	A GVD EQU
00152		0008	A CFH EQU
00153			*
00154		0010	A PFSZ EQU
00155			*
00157			* VARIABLES
00158			*
00159A D000			* ORG
00160		D000	A VAREL EQU
00161A D000		0001	A ..ABRT RMB
00162A D001		0001	A ..CONT RMB
00163			*
00164			* KEYBOARD
00165			*
00166A D002		0001	A KLOCK RMB
00167A D003		0001	A KBEFLG RMB
00168A D004		0001	A KCOUNT RMB
00169A D005		0002	A KHEAD RMB
00170A D007		0002	A KTAIL RMB
00171			*
00172			* TIMER
00173			*
00174A D009		0001	A ..PFIQ RMB
00175A D00A		0001	A ..THAC RMB
00176A D00B		0001	A ..STC RMB
00177A D00C		0004	A ..\$T1 RMB

```

000081          STA INSLD
000082          ENDM
000083          SEF   MACR
000084          ORCC #$40
000085          ENDM
000086          CLF   MACR
000087          ANDCC #$BF
000088          ENDM
000089          +
000090          +
000091          +
000092          +
000093          +
000094          0000   A   ..VB   EQU   $0000   BLUE VPAN
000095          4000   A   ..VR   EQU   $4000   RED VPAN
000096          8000   A   ..VG   EQU   $8000   GREEN VPAN
000097          C000   A   ..CRB  EQU   $C000   CHARACTER BUFFER
000098          C7D0   A   ..ATB  EQU   ..CFB+$2000 ATTRIBUTE BUFFER
000099          D380   A   .COMAR EQU   $D380   COMMUNICATION AREA
000100          D300   A   ..CENT EQU   $D300   ORIGINAL FONT
000101          D000   A   STX TOP EQU   $D000
000102          +
000103          + COMMUNICATION REG.
000104          +
000105          D380   A   EFPPG EQU   .COMAR ERROR CODE
000106          D381   A   STATUS EQU   .COMAR+1
000107          D382   A   CHDPE EQU   .COMAR+2 COMMAND CODE
000108          D383   A   CHDPM EQU   .COMAR+3 PARAMETER
000109          D400   A   CMDEN EQU   .COMAR+128 END OF AREA PAST 1
000110          +
000111          00F0   A   PF1   EQU   $0F   STATUS 0-3
000112          0010   A   TM1   EQU   $10   STATUS 4
000113          0020   A   IN1   EQU   $20   STATUS 5
000114          0040   A   NWD1  EQU   $40   STATUS 6
000115          0080   A   CNTFLG EQU   $80   STATUS 7
000116          00F0   A   PF1$  EQU   -PF1-$FF01
000117          00EF   A   TM1$   EQU   -TM1-$FF01
000118          00DF   A   IN1$   EQU   -IN1-$FF01
000119          00EF   A   NWD1$  EQU   -NWD1-$FF01
000120          007F   A   CNTFL$ EQU   -CNTFLG-$FF01

```

00178A	D010	0004	A	.\$T11	RME
00179A	D014	0004	A	.\$T2	RME
00180A	D018	0004	A	.\$T20	RME
00181A	D012	0001	A	.\$CURS	RME
00182A	D01C	0001	A	.\$CURSM	RME
00183A	D010	0001	A	.\$CURTM	RME
00184A	D01E	0001	A	.\$CURST	RME
00185A	D01F	0002	A	.\$VOFST	RME
00186A	D021	0001	A	.\$CSTGL	RME
00187A	D022	0001	A	.\$PUTCFG	RME
00188A	D023	0001	A	.\$CLSCOS	RME
00189A	D024	000A	A	.\$TABST	RME
00190A	D02E	0001	A	.\$ACL	RME
00191A	D02F	0001	A	.\$ECL	RME
00192A	D030	0001	A	.\$ECLX	RME
00193A	D031	0001	A	.\$CWD	RME
00194A	D032	0001	A	.\$CHT	RME
00195A	D033	0001	A	.\$TNS	RME
00196A	D034	0001	A	.\$EKEY	RME
00197A	D035	0001	A	.\$LNC0	RME
00198A	D036	0001	A	.\$LND	RME
00199A	D037	0001	A	.\$DNL	RME
00200A	D038	0001	A	.\$FS	RME
00201A	D039	0002	A	.\$VCU	RME
00202A	D03B	0002	A	.\$U	RME
00203A	D03C	0008	A	.\$DMAP	RME
00204A	D045	0001	A	.\$BPY	RME
00205A	D046	0001	A	.\$BPX	RME
00206A	D047	0002	A	.\$BPAD	RME
00207A	D049	0002	A	.\$BAAD	RME
00208A	D048	0002	A	.\$BAD	RME
00209A	D04D	0003	A	.\$MDRG	RME
00210A	D050	0002	A	.\$GTCP	RME
00211A	D052	0002	A	.\$GTCP	RME
00212A	D054	0002	A	.\$GTPOA	RME
00213A	D056	0002	A	.\$GETNA	RME
00214A	D058	0001	A	.\$CY	RME
00215A	D059	0001	A	.\$CX	RME
00216A	D05A	0002	A	.\$CEAD	RME

```

$0400
$0400
ICOD1,1 MSB I/O
,$ORG
,$ORG+1
,ICOD2,1 I/O-KD7
,$ORG
,$ORG+1
ACKIFQ,1 ACKNOWLEDGE IFQ
,$ORG
,$ORG+1
BUZZEF,1 BUZZER
,$ORG
,$ORG+1
ATTENT,1 MAIN SYSTEM REQUEST
,$ORG
,$ORG+1
IORDUM,3 DUMMY
,$ORG
,$ORG+3
CRTON,1 DISPLAY ON/OFF
,$ORG
,$ORG+1
VACC,1 VRAM ACCESS
,$ORG
,$ORG+1
BUSY,1 BUSY STATE
,$ORG
,$ORG+1
IORDM2,2 DUMMY
,$ORG
,$ORG+2
INSLED,1 INSERT MODE LED
,$ORG
,$ORG+1
SOFFST,2 VRAM OFFSET REG.
,$ORG
,$ORG+2

```

```

$07 COLOR MASK ON ATTR.
$08 REVERSE MASK ON ATTR.
$10 PROTECT MASK ON ATTR.
$40 MODIFIED TAG MASH ON ATTR.
$80 FIELD START MASH ON ATTR.

80 NUMBER OF COLUMN IN LINE,MANY
40 NUMBER OF COLUMN IN LINE,LITTLE
25 NUMBER OF LINE,MANY
20 NUMBER OF LINE,LITTLE
80 HORIZONTAL BYTES,GRAPHIC
640 HORIZONTAL DOT
200 VERTICAL DOT
8 HEIGHT OF CHAR FONT

```

```

$0000
#
1 ABORT FLAG
1 CONT FLAG SAVE

1 KEY LOCK/UNLOCK
1 BUFFER MODE FLAG
1 NUMBER OF CODE IN KEY BUFFER
2 ADDRESS OF CURRENT BUFFER HEAD
2 ADDRESS OF CURRENT BUFFER TAIL

1 PF INTERRUPT CODE
1 ACCESS FLAG
1 CONTROL
4 CLOCK

```

```

4 INTERRUPT TIME
4 COUNTER
4 RESTORE DATA

1 STATE REG. OF CURSOR DISPLAY
1 TIMER OF CURSOR

2 VRAM OFFSET FOR SCROLL
1
1 CONTINUANCE PUT FLAG.
1 CONSOLE COLOR DISABLE(MONO SELECTION)

10
1 ATTR COLOR CODE
1 BACK COLOR CODE

1 CHAR WIDTH FLAG,0=SMALL,1=LARGE
1 CHAR UNDER MARGIN,0 OR 2.
1 INSERT FLAG
1 ENTER KEY CODE
1 0 LONG
1 NUMBER OF COLUMNS IN LINE
1 NUMBER OF LINES IN DISPLAY
1 KEY INDICATE FLAG
1 VRAM ADDR UNIT OF CHR COLUMN
1 VRAM ADDR UNIT OF CHR LINE
1 MAP OF DISPLAY PLACE
1 BUFFER PTR X
1 BUFFER PTR Y
1 BUFFER PTR (ADDR)
1 BUFFER PTR,ATTR.
1 BUFFER PTR,VRAM ADDR
1 CURR. DISPLAY WINDOW
1 CURRENT WINDOW TOP (CHAR)
1 CURRENT WINDOW BOTTOM (CHAR)
1 CURRENT WINDOW TOP (ATTR)
1 CURRENT WINDOW BOTTOM (ATTR)
1 CURSOR ADDR: Y
1 CURSOR ADDR: X
1 CURSOR BUFFER ADDR

```

```

00217A D05C 0002 A .CAAD RMB 12 CURSOR ATTR ADDR
00218A D05E 0002 A .CVAD RMB 12 CURSOR VRAM ADDR
00219A D060 0001 A .3CL RMB 1 COLOR CODE
00220A D061 0001 A .BF RMB 1 FUNCTION CODE
00221A D062 0001 A .3Y RMB 12 AXIS X GRAPH
00222A D064 0002 A .GA RMB 12 AXIS Y GRAPH
00223A D066 0002 A .GAD RMB 12 VRAM ADDR
00224A D068 0001 A .BEM RMB 12 BIT MAP
00225A D069 0002 A .80XX1 RMB 12 BOX X1
00226A D06E 0002 A .80X11 RMB 12 BOX Y1
00227A D06D 0002 A .80X2 RMB 12 BOX Y2
00228A D06F 0002 A .80Y2 RMB 12 BOX Y2
00229A D071 0002 A .LX1 RMB 12 X CO-ORDINATE OF POINT1
00230A D073 0002 A .LY1 RMB 12 Y CO-ORDINATE OF POINT1
00231A D075 0002 A .LX2 RMB 12 X CO-ORDINATE OF POINT2
00232A D077 0002 A .LY2 RMB 12 Y CO-ORDINATE OF POINT2
00233A D079 0002 A .CBTOP RMB 12 CLEAR BUFFER TOP ADDR
00234A D07E 0002 A .CBBTM RMB 12 CLEAR BUFFER BOTTOM ADDR+1
00235A D07D 0002 A .CVTOP RMB 12 CLEAR VRAM TOP ADDR
00236A D07F 0002 A .CVBTM RMB 12 CLEAR VRAM BOTTOM ADDR+1
00237A D081 0002 A .ERSTX RMB 12 ERASE IX
00238
00239A D083 0079 A .%ORG SOFG .CBTOP
00240A D083 0079 A .%ORG EQU .CBTOP
00241A D083 007C A .BLUSD EQU DS ELUSD,3 LINEX SIDE. BLUE
00242A D083 007F A .%ORG SET .%ORG+3
00243 D082 A .FEDSD EQU DS FEDSD,3 LINEX SIDE. RED
00244A D083 007F A .%ORG SET .%ORG+3
00245A D083 007F A .GRESID EQU DS GRESID,3 LINEX SIDE. GREEN
00246A D083 0082 A .%ORG SET .%ORG+3
00247A D083 0012 A .EXFONT RMB 18 EXT D FONT
00248A D083 0083 A .%ORG SOFG EXFONT
00249A D083 0083 A .%ORG SET EXFONT
00250A D083 0083 A .%ORG EQU ELUSD,3 RAM HOOK BLUE DOT DISPLAY
00251A D083 0086 A .%ORG SET .%ORG+3

```

```

00247A D083 0086 A .%ORG DS FEDDSP,3 RAM HOOK RED DOT DISPLAY
00248A D083 0089 A .%ORG SET .%ORG+3
00249A D083 0089 A .GRESID EQU DS GRESID,3 RAM HOOK FOR GREEN DOT DISPLAY
00250A D083 0089 A .%ORG SET .%ORG+3
00251A D083 0089 A .%ORG DS ELUSD,3 RAM HOOK FOR X PARALLEL LINE'S BLUE
00252A D083 0089 A .%ORG SET .%ORG+3
00253A D083 0089 A .%ORG DS FEDLIN,3 RAM HOOK FOR X PARALLEL LINE'S RED
00254A D083 0089 A .%ORG SET .%ORG+3
00255A D083 0089 A .%ORG DS GRELIN,3 RAM HOOK FOR X PARALLEL LINE'S GREEN
00256A D083 0089 A .%ORG SET .%ORG+3
00257A D083 0089 A .%ORG DS .%ORG+3
00258A D083 0089 A .%ORG SET .%ORG+3
00259A D083 0089 A .%ORG DS .%ORG+3
00260A D083 0089 A .%ORG SET .%ORG+3
00261A D083 0089 A .%ORG DS .%ORG+3
00262A D083 0089 A .%ORG SET .%ORG+3
00263A D083 0089 A .%ORG DS .%ORG+3
00264A D083 0089 A .%ORG SET .%ORG+3
00265A D083 0089 A .%ORG DS .%ORG+3
00266A D083 0089 A .%ORG SET .%ORG+3
00267A D083 0089 A .%ORG DS .%ORG+3
00268A D083 0089 A .%ORG SET .%ORG+3
00269A D083 0089 A .%ORG DS .%ORG+3
00270A D083 0089 A .%ORG SET .%ORG+3
00271A D083 0089 A .%ORG DS .%ORG+3
00272A D083 0089 A .%ORG SET .%ORG+3
00273A D083 0089 A .%ORG DS .%ORG+3
00274A D083 0089 A .%ORG SET .%ORG+3
00275A D083 0089 A .%ORG DS .%ORG+3
00276A D083 0089 A .%ORG SET .%ORG+3

```

```

00277A E00A 80 81 A REST00 STD .X++
00278A E00C 8C D360 A .CMPX BLO #VAREL$
00279A E00F 29 F3 EQUA .BLO REST00
00280A E011 8C D360 A .LDX #IYBUF KEY BUFFER PTR INITIALIZE
00281A E014 9F 05 A .STX I.HEAD
00282A E016 9F 07 A .STX I.TAIL THAT IS CIPCLAR.
00283A E018 9F 07 A .LDA #S07
00284A E01A 97 2F A .STA .I.ACL ATTRIBUTE
00285A E01C FC D400 A .LDD I.COD1 TURN OFF THE KEY STROBE.
00286A E01F 86 FF A .LDA .SFF SET BUFFER MODE ON
00287A E021 97 03 A .STA I.BFFLG
00288
00289A E023 10CE D000 A .LDS #STI TOP INIT SP
00290
00291A E027 8D E1CB A .JSP DEFALT DEFAULT INIT
00292A E02A 86 23 A .LDA #S23
00293A E02C 8D EDCB A .JSP CNSINT
00294
00295A E02F 7F D381 A .CLP STATUS CLEAR INTERRUPT
00296A E032 1C AF A .ANDCC #SAF REMOVE IFQ AND FIQ MASK BITS
00297
00298
00299
00300
00301
00302A E034 10CE D000 A .LDS #STI TOP INIT SP
00303A E038 7F D382 A .CLP CMDREG COMMAND NON
00304A E03B 8D E037 A .JSP CURGO
00305A E03E 8D E133 A .JSP CMDWA1
00306
00307
00308A E041 34 02 A .CMOCT1 EQU 1
00309A E043 8D E001 A .PSHS A
00310A E046 8D E12A A .JSP CURSTP
00311A E049 35 02 A .CMOCT2 JSP COMINT CTL REG CLEAR
00312A E04B 9F 00 A .PULS A
00313A E04D F6 E076 A .CLR .I.ABRT CLEAR ABORT FLAG
00314A E050 8E E077 A .LDB CMDPN NO. OF AVAILABLE COMMAND
00315
00316A E053 A1 84 A .LDX .I.CMDPTR COMMAND ENTRY TABLE

```

# FM-7 サブシステム・モニタ ソースリスト

```

00317A E055 25 04 E05B BLO CMOCT3 LOW
00318A E057 A1 01 A CMA 1,X
00319A E059 23 08 E063 BLS CMOJMP EXIST THIS BLOCK
00320A E061 30 04 A CMOCT3 LEAX NEXT BLOCK
00321A E05D 5A 00 A CMOCT3 DEC8
00322A E05E 26 F3 E053 CMOCT2
00323A E060 7E E0ED A JMP ERR70 UNAVAILABLE
00324
00325 E063 A CMOJMP EQU #
00326A E063 A0 84 A SUBA 0,X OFFSET ON BLOCK
00327A E065 48 00 A LSLA
00328A E066 AE 00 A LDX 2,X VECTOR TABLE TOP
00329A E066 AE 86 A LDX A,X
00330
00331A E064 A0 84 A GOCMD JDF X BRANCH TO COMMAND
00332
00333A E06C 20 C6 E034 CMSEND BFA CMOCTL SEND & WAIT
00334
00335 E06E A IRQAE EQU #
00336A E06E A01 ACI
00337A E071 86 D402 A BITA ACKIFQ
00338A E073 97 FF A LDA #FFF SET ABOPT FLAG
00339A E075 38 00 A STA ..ABRT
00340
00341A E076 04 A CMOPI FCB (CON%-CMOPIF) / 4
00342A E077 0100 A CMOPIF $0100,CON% CONSOLE BLOCK
00343A E078 1520 A FDB $1520,GR% GRAPHIC BLOCK
00344A E07D E0A1 A FDB $E0C0,KEY% KEYBOARD BLOCK
00345A E083 E069 A FDB $303F,TIM% TIMER BLOCK
00346A E086 E0C1 A FDB
00347A E087 E14E A CON% FDB INIT
00348A E089 E24E A FDB ERASE
00349A E08B E5E1 A FDB PUT
00349A E08D E5E0 A FDB GET
00350A E08F E546 A FDB GETC
00351A E091 ECF5 A FDB CGEL1

```

```

00352A E093 E072 A FDB CPBL1
00353A E096 ECF4 A FDB CPBL2
00354A E097 E071 A FDB CPBL3
00355A E099 ECEC A FDB GETBAD
00356A E09B E0C8 A FDB TABSET
00357A E09D E0C8 A FDB CMCCTL
00358A E09F E241 A FDB NERASE
00359A E0A1 E088 A FDB LINE
00360A E0A3 F0C4 A FDB CHAIN
00361A E0A5 F412 A FDB POINT
00362A E0A7 F48A A FDB PAINT
00363A E0A9 F103 A FDB SYMBOL
00364A E0AB F216 A FDB CHANGE
00365A E0AD F286 A FDB GBLK1
00366A E0AF F446 A FDB GBLK2
00367A E0B1 F386 A FDB GBLK3
00368A E0B3 F589 A FDB GBLK4
00369A E0B5 F670 A FDB GBLK5
00370A E0B7 E072 A FDB CHRLN
00371A E0B9 F05C A KEY% FDB INKEY
00372A E0BB FE02 A FDB FUNKEY
00373A E0BD FE46 A FDB GSTRPF
00374A E0BF FE51 A FDB PFINT
00375A E0C1 FE7C A TIM% FDB SETTIM
00376A E0C3 FE44 A FDB TOTIM
00377A E0C5 FF6F A FDB TEST
00378
00379A E0C7 00 A INPARG FCB 0,80,25,0,25,0,1,0
00380 A E0C8 50 A
00381 A E0C9 19 A
00382 A E0CA 00 A
00383 A E0CB 19 A
00384 A E0CC 00 A
00385 A E0CD 01 A
00386 A E0CE 00 A
00387
00388
00389
00390
00391
00392
00393
00394
00395
00396
00397
00398
00399
00400
00401
00402
00403
00404
00405
00406
00407
00408A E0EF E7 Q380 A ERROR EQU #
00409A E0F2 7E E06C A STA COMAR SET ERROR STATUS
00410
00411
00412
00413 E0F5 A CONTO EQU #
00414

```

```

00394
00395A E0CF 86 3C A ERR50 LDA #50 INIT PARAM
00396A E0D1 8C A FCB #50 CMPX #
00397A E0D2 86 3D A ERR51 LDA #51 CO-ORDINATE OF CONSOLE
00398A E0D4 8C A FCB #52 CMPX #
00399A E0D6 3E A ERR52 LDA #52 ORDER SEQ.
00400A E0D8 8C A FCB #53 CMPX #
00401A E0DA 3F A ERR53 LDA #53 CO-ORDINATE OF GRAPHIC
00402A E0DC 8C A FCB #54 CMPX #
00403A E0DE 40 A ERR54 LDA #54 FUNCTION CODE
00404A E0E0 8C A FCB #55 CMPX #
00405A E0E2 41 A ERR55 LDA #55 NO. OF CO-ORDINATE
00406A E0E4 8C A FCB #56 CMPX #
00407A E0E6 42 A ERR56 LDA #56 NO. OF CHARACTER
00408A E0E8 8C A FCB #57 CMPX #
00409A E0EA 43 A ERR57 LDA #57 NO. OF COLOR
00410A E0EC 8C A FCB #58 CMPX #
00411A E0EE 44 A ERR58 LDA #58 PF NO.
00412A E0F0 8C A FCB #59 CMPX #
00413A E0F2 45 A ERR59 LDA #59 INVALID PARAMETER
00414A E0F4 8C A FCB #60 CMPX #
00415A E0F6 46 A ERR60 LDA #60 INVALID COMMAND
00416A E0F8 8C A FCB #61 CMPX #
00417A E0FA 47 A ERR61 LDA #61 ERROR EQU #
00418A E0FC 8C A STA COMAR SET ERROR STATUS
00419A E0FE 7E E06C A JMP CMSEND SEND
00420
00421
00422
00423
00424
00425
00426
00427
00428
00429
00430
00431
00432
00433
00434
00435
00436
00437
00438
00439
00440
00441
00442
00443
00444
00445
00446
00447
00448
00449
00450
00451
00452
00453
00454

```

```

00415 # INTERMEDIATE
00416
00417 # CONTINUOUS
00418 # BUSY LINE
00419 # AND WAIT
00420
00421 # IF NEW
00422 # BREAK NOW
00423
00424A E0F5 34 02 A PSHS
00425A E0F7 36 80 A LDA
00426A E0F9 8A D381 A DRA
00427A E0FB 87 D381 A STA
00428A E0FD 8D 32 E133 A BSR
00429A E101 81 64 A CMA
00430A E103 26 10 E115 A ENE
00431
00432 #
00433A E105 7F D380 A CONTO2 EQU
00434A E108 86 7F A CLR
00435A E10A 84 D381 A LDA
00436A E10D 87 D381 A ANDA
00437A E110 7F D382 A STA
00438A E113 82 A RTS
00439
00440 #
00441A E115 10CE D009 A CNTABT LDS
00442A E119 34 00 A PSHS
00443A E11B 6D EDE8 A JSR
00444A E11E 7E E046 A JMP
00445
00446 #
00447 #
00448 # GET CONTINUOUS
00449
00450 #
00451 # BUSY LINE
00452 # WAIT FOR
00453 # AND SAVE
00454
00455 # IF NEW

```

```

00455 # BREAK NOW
00456
00457A E121 34 02 A PSHS
00458A E123 8D 0E E133 A BSR
00459A E125 81 64 A CMA
00460A E127 26 EC E115 A ENE
00461A E129 8C A SKIP2
00462
00463A E12A 34 02 A COMINT PSHS
00464A E12C 86 D381 A CONTI2 LDA
00465A E12F 97 01 A STA
00466A E131 20 D2 E105 A BRA
00467
00468 #
00469 #
00470A E133 86 E133 A CMDWA1 EQU
00471A E135 84 7F A CMDWA5 LDA
00472A E137 87 D380 A ANDA
00473A E139 87 D380 A STA
00474A E13B 87 D380 A BSYOFF
00475A E13E 7D D40A A TST
00476A E140 86 D382 A CMDWA6 LDA
00477A E142 26 07 E14A A ENE
00478A E144 86 D380 A LDA
00479A E146 26 E8 E133 A BSI
00480A E148 20 F4 E13E A BRA
00481A E14A 87 D40A A CMDWA9 BSYON
00482A E14C 87 D40A A STA
00483A E14E 39 RTS
00484
00485 #
00486 #
00487 #
00488 #
00489 #
00490 #
00491 #
00492 #

```

```

00493 #
00494 #
00495 #
00496 #
00497A E14E 0F 22 A INIT EQU
00498A E150 8D EDEF A CLR
00499A E153 CE D380 A JSR
00500A E155 A8 43 A LDA
00501A E157 37 2F A STA
00502A E159 5F 00 A CLRB
00503A E15B 8E 0001 A LDX
00504A E15E A6 44 A LDA
00505A E160 81 50 A CMA
00506A E162 27 08 E16C A BEQ
00507A E164 61 28 A CMA
00508A E166 28 5E E1C6 A ENE
00509A E168 C6 0E A LDB
00510A E16A 30 01 A LEAX
00511A E16C 37 36 A INIT21 STA
00512A E16E 07 31 A STB
00513A E170 8F 39 A STX
00514A E172 8F 39 A CLRB
00515A E173 8E 0280 A LDX
00516A E175 A6 45 A LDA
00517A E178 81 19 A CMA
00518A E17A 27 09 E185 A BEQ
00519A E17C 81 14 A CMA
00520A E17E 26 46 E1C6 A ENE
00521A E180 C6 02 A LDB
00522A E182 8E 0320 A LDX
00523A E185 97 37 A INIT31 STA
00524A E187 07 32 A STB
00525A E189 3F 38 A STX
00526A E18B 3F 38 A LDA
00527A E18D 26 46 A LDB
00528A E18F 07 35 A STB
00529A E191 27 02 E195 A BEQ
00530A E193 4A 4A A DECA
00531A E194 4A 4A A DECA
00532A E195 97 43 A INIT40 STA

```

## SEND PROCESS

FLAG ON,  
LOW,  
FOR CONTINUE COMMAND.

COMMAND IS NOT CONTINUE,  
COMMAND AND DO NEW COMMAND.

```

A
#CNTFLG
STATUS
STATUS
CMDWAI WAIT
#100 CONTINUE ?
CNTABT

*
ERRREG
#CNTFLG
STATUS
STATUS
CMDREG
A
A,PC

*****
#STH TOP
A
CURSET
CMDCNT

```

## DATA

LOW,  
CONTINUE COMMAND,  
CONT FLAG.

COMMAND IS NOT CONTINUE,

COMMAND AND DO NEW COMMAND.

```

A
CMDWAI WAIT
#100 CONTINUE ?
CNTABT

$BC CMPX #

A
STATUS
..CONT
CNT02

*****
* WAIT FOR COMMAND
ERRREG
#$F
ERRREG

BUSY
CMDREG
CMDWAS
ERRREG
CMDWAS
CMDWAS
CMDWAS
BUSY

*****
INIT - INITIALIZE CODE -

BACK COLOR CODE
NO. OF COLUMN ON LINE
NO. OF LINE ON DISPLAY
START LINE NO. OF SCROLL

```

NO. OF SCROLL LINE  
INDICATE FLAG

```

*
PUTCFG
CURCLR
#.COMAP
3,U
..BCL SET BACK COLOR

#1
4,U
#CSCX1
INIT21
#CSCX2
INIT90 WIDTH ERROR
#8 CHAR WIDTH FLAG=8=LARGE
1,X
..LNC
..CWD CHAR WIDTH FLAG SET
..VCU UNIT OF STEP

#GXSZ(CFH)
5,U SET NO. OF LINE
#CSCY1
INIT31
#CSCY2
INIT90 HEIGHT ERROR
#8 CHAR HEIGHT FLAG=2=UNDER MARGIN
#GXSZ(CFH+2)
..DNL
..CHT CHAR HEIGHT FLAG SET
..VLU UNIT OF LINE STEP
HFLG
8,U
..FS INDICATE FLAG
INIT40

USE 2 LINE FOR INDICATE
..DMAP+2K3 INDICATE START LINE

```

```

00533A E197 37 42 A STA ..DMAP+2+1 PAGE2 END LINE
00534A E199 41 46 A CMPA 6,U
00535A E19E 23 29 E1C6 BLS INIT90 SCROLL START ERROR
00536A E19D A6 46 A LDA 6,U
00537A E19F 37 3D A STA ..DMAP+2+0 SCROLL START LINE
00538A E1A1 37 40 A STA ..DMAP+2+1 PAGE1 END LINE
00539A E1A3 A8 47 A ADDA 7,U
00540A E1A5 31 43 A CMPA ..DMAP+2+3
00541A E1A7 22 1D E1C6 BHI INIT90
00542A E1A9 37 3E A STA ..DMAP+2+0+1 SCROLL END
00543A E1AE 37 41 A STA ..DMAP+2+2 PAGE2 START
00544A E1AD 0F 3F A CLF ..DMAP+2+1 PAGE1 START 0,CONSTANT
00545A E1AF 36 37 A LDA ..DNL
00546A E1B1 37 44 A STA ..DMAP+2+3+1 INDICATE END,DISPLAY END
00547A E1B3 6D 47 A TST 7,U
00548A E1B5 28 0F E1C6 BMI INIT90 SCROLL NEG.
00549A E1B7 A6 4A A LDA 10,U SET CONSOLE COLOR: DISABLE SWITCH
00550A E1B9 27 02 E1BD BEQ INIT50
00551A E1BB 86 FF A LDA #FFF
00552A E1BD 37 23 A INIT50 STA CSCLCDS
00553A E1BF A6 49 A LDA 9,U
00554A E1C1 27 17 E1DA BEQ HOME NOT ERASE
00555A E1C3 7E E25E A JMP ERSALL ERASE
00556
00557A E1C5 86 3C A INIT90 LDA #60 ERROR CODE
00558A E1C7 67 D380 A STA STA
00559A E1C9 3E EDC7 A DEFALT LDX #INPARM
00560A E1CE 103E D383 A LDY #CMOPFM
00561A E1D2 06 08 A LDB #8
00562A E1D4 6D E44A A JSR MOVE
00563A E1D7 7E E14E A JMP INIT DEFAULT SET
00564
00565
00566
00567 E1DA A HOME EQU #
00568A E1DA 8D E60E A JSR INDICT
00569A E1DD CE D03D A LDA #..DMAP
00570A E1E0 5F CLRE
00571A E1E1 5C HOME10 INCE
00572A E1E2 A6 C0 A LDA ,U+

```

```

00573A E1E4 A1 C0 A CMPA ,U+
00574A E1E6 27 F9 E1E1 BEQ HOME10
00575A E1E8 37 4D A STA .MORG
00576A E1EA 37 45 A STA .BPY
00577A E1EC D7 4F A STB .MORG+2
00578A E1EE E6 5F A LDB -1,U
00579A E1F0 D7 4E A STB .MORG+1
00580A E1F2 8D E375 A CHGWIN JSR GETRNG
00581A E1F5 8D 03 E1FA BSR BPTLRL
00582A E1F7 7E EDE8 A JMP CURSET
00583A E1FA 0F 46 A BPTLRL CLP
00584A E1FC 8D 02 E200 BUFPTR BSR EADDR
00585A E1FE 20 1C E21C BSR EADDR
00586
00587 E200 A BADDR EQU #
00588A E200 8D 08 E20A BSR BADPN
00589A E202 3F 47 A STX .EPAD
00590A E204 109F 49 A STY .EADD
00591A E207 39 A RTS
00592
00593A E208 5F BADPL CLRE
00594A E209 SI IP2
00595A E20A DC 8C A FCB $BC CMPX #
00596A E20B 0C 45 A BADPN LDD .BPY
00597A E20D 34 04 A BADR EQU #
00598A E20E D6 36 A PSHS B
00599A E210 30 60 A LDB ..LNC
00600A E211 E8 E0 A MUL
00601A E213 89 C0 A ADDB ,S+
00602A E215 1F C0 A ADCA ..CR6/256
00603A E217 31 01 A TFR D,X
00604A E218 39 89 07D0 A LEAY ..ATB-..CR6,X
00605
00606 E21C A EVADR EQU #
00607A E21C 8D 05 E223 BSR EVADN
00608A E21E D0 48 A STD .EVAD
00609A E220 39 A RTS
00610

```

```

00611A E221 5F BVADL CLRE
00612A E222 SI IP2
00613A E223 DC 3C A FCB $BC CMPX #
00614 E225 A EVADN LDD .BPY
00615A E226 34 02 A EVAD EQU #
00616A E227 96 3A A PSHS A
00617A E229 3D 04 A LDA ..VCU+1
00618A E22A 34 06 A MUL
00619A E22C A6 62 A PSHS D
00620A E22E D6 3C A LDA 2,S
00621A E230 3D 04 A LDB ..VLU+1 LSB
00622A E231 E3 E4 A ADDD ,S
00623A E233 E0 E4 A STD ,S
00624A E235 A6 3E A LDA 2,S
00625A E237 D6 38 A LDB ..VLU MSE
00626A E239 3D 04 A MUL
00627A E23A 1E 89 A EYB A,B
00628A E23C E3 E4 A ADDD ,S
00629A E23E 32 63 A LEAS 3,S
00630A E240 39 A RTS
00632
00633 * ERASE COMMAND
00634
00635 *
00636 * .COMAR+3 WINDOW NO.
00637 * .COMAR+4 BACK COLOR CODE
00638 * .COMAR+5 COLOR CODE (NEW ERASE COMMAND)
00639
00639A E241 0F 22 A NERASE CLR
00640A E243 8D EDEF A JSR PUTCFG
00641A E246 F6 D385 A LDB CURCLR
00642A E249 20 07 E252 A LDB .COMAR+5 LOAD COLOR.
00643
00644 E24B A ERASE EQU #
00645A E248 0F 22 A CLR
00646A E24D 8D EDEF A JSR CURCLR
00647A E250 D0 2E A LDB .ACL
00648 E252 A ERASDD EQU #
00649A E252 86 D383 A LDA .COMAR+3
00650A E255 26 41 E298 BNE ERASE0

```



# FM-7 サブシステム・モニタ ソース・リスト

```

00651A E257 D7 2E A STR ..ACL SET THE COLOR CODE.
00652A E259 B6 D384 A LDA .COMAR+4
00653A E25C 97 2F A STA .BCL SET BACK COLOR
00654 E25E A ERSALL EQU #
00655A E25E 8E 0000 A LDX #0
00656A E261 CE 4000 A LDU #4000
00657A E264 00 23 A NEG CCLCDS
00658A E266 SCROFF
00659A E269 BD D408 A STA CRTON
00660A E26C 8D E2F4 A JSP ERSS04
00661A E26C 65 D408 A SCRON
00662A E271 96 3F A BITA CRTON
00663A E273 D6 44 A NEG CCLCDS
00664A E275 00 70 A LDA ..DMAP+2 TOP VRAM Y
00665A E277 BD E331 A LDB ..DMAP+24-1 BOTTOM VRAM Y PAST 1
00666A E27A BD E10A A STD .CVTOP
00667A E27D CE D13D A JSP ERSS0F
00668A E280 A6 C1 A ASSFLO LDU #..DMAP ASSUME FIELD MARK
00669A E282 BD E208 A ERSALB LDA .U+
00670A E285 A6 A4 A JSP BADPL
00671A E287 8A 80 A LDA .Y
00672A E289 A7 A4 A ORA #AMFL
00673A E28B 1183 D045 A STA .Y
00674A E28F 22 06 E287 A CMPLU #..DMAP+8
00675A E291 25 ED E287 A ERSAL9 ERSAL9
00676A E293 33 5F A BLD ERSAL5
00677A E295 20 E9 E280 A LEAU -1,U
00678A E297 39 ERSAL3 RTS
00679 E299 A ERSAL3
00680A E29B 31 03 A ERASE0 CMA #3
00681A E29A 1022 FE31 E0CF LBLH ERK60
00682A E29E D7 2E A STB .ACL SET THE COLOR CODE.
00683 E2A0 A ERASE1 EQU #
00684A E2A0 48 LSLA
00685A E2A1 8E D08B A LDX #..DMAP-2
00686A E2A4 30 86 A LEAX A,X
00687A E2A6 E6 84 A LDE O,X
00688A E2A8 E1 01 A CMPE 1,X

```

```

00689A E2AA 1027 FE21 E0CF LBEQ ERK60
00690A E2AC 44 LSLA
00691A E2AF 37 4F A STA .MDRG+2 I IND
00692A E2B1 EC 84 A LDD .X
00693A E2B3 40 TSTA IS THE CURRENT WINDOW
00694A E2B4 26 04 E2BA BNE ERASE5 FULL OF LOGICAL DISPLAY AREA?
00695A E2B6 01 43 A CMPE #..DMAP+213
00696A E2B8 27 A4 E2BE BEQ ERSALL YES.
00697 E2BA A ERASE5 EQU #
00698A E2BA DD 70 A STD .CVTOP
00699A E2BC 4D A4 A STD .MDRG
00700A E2BE 97 45 A STA BPY
00701A E2C0 BD E1F2 A JSP CHGWIN
00702A E2C3 8D 60 E325 ESR ERSS0F
00703A E2C5 36 40 A ASMFLD LDA .MDRG ASSUME FIELD TAG AT CURRENT WINDOW TOP.
00704A E2C7 BD E208 A JSP BADPL
00705A E2CA A6 A4 A LDA .Y
00706A E2CC 34 80 A ORA #AMFL
00707A E2CE A7 A4 A STA .Y
00708A E2D0 7E E1FC A JMP BUPTIF
00709 E2D2 A
00710 E2D4 A
00711 E2D6 A
00712 E2D8 A ERASE DISPLAY, CHARACTER BUFFER
00713 E2DA A AND ATTRIBUTE BUFFER
00714 E2DC A
00715 E2DE A
00716 E2E0 A CVTOP START LINE OF ERASE
00717 E2E2 A CVTOP+1 PAST END LINE OF ERASE
00718 E2E4 A
00719 E2E6 A
00720A E2D3 96 E2D3 A ERSVRM EQU #
00721A E2D5 BD E211 A LDA .CVTOP+1 GET ADDRESS THAT IS PAST 1
00722A E2D7 1F 04 A JSP BVALD TO END OF ERASE AREA.
00723A E2D9 83 4000 A D.U.
00724A E2DD 23 0E E2ED SUD #4000 OUT OF BLOCK?
00725A E2DF CE 4000 A BLS ERSS03 NO.
00726A E2E1 34 06 A LDU #4000
00727A E2E4 8D 07 E2ED PSHS D
00728A E2E6 8E 0000 A BSR ERSS03
00729A E2E8 35 40 A LDX #0
00730A E2EB 20 07 E2F4 A PULS U
00731 E2ED A ERSS03 EQU #
00732A E2ED 96 70 A .CVTOP GET TOP ADDRESS OF ERASE.
00733A E2EF BD E221 A JSP BVALD
00734A E2F2 1F 01 A TFR D,X SET X.
00735A E2F4 96 23 A ERSS04 LDA CCLCDS COLOR ENABLE?
00736A E2F6 27 1F E317 BEQ EPSVN YES.
00737A E2F8 26 06 E305 BNE EPSMND NO, WILL I ERASE THE ONLY GREEN?
00738A E2FA 86 04 A LDB #4
00739A E2FC D6 2E A LDB ..ACL
00740A E2FE C5 08 A BITB #AMPV REVERSE?
00741A E300 26 10 E31F BNE ERSS05 YES, DISPLAY IS ERASED BY GREEN.
00742A E302 4F 1A E31F CLPA
00743A E303 20 89 E30A A ERSS05 DISPLAY IS ERASED BY BLACK.
00744A E305 80 C8 E300 A .V.G,X I WILL ERASE ONLY GREEN VRAM.
00745A E307 33 C8 E300 A LEAU ..V.G,U
00746A E30D 4F C3 A CLPA
00747A E30E D6 2E A LDB ..ACL
00748A E310 C5 08 A BITB #AMPV
00749A E312 27 54 E368 BEQ ERSV
00750A E314 4C 51 E368 INCA
00751A E316 20 5E E368 ERSV
00752A E317 96 21 A ERSVN LDA ..ACL
00753A E319 85 08 A BITA #AMPV
00754A E31B 26 02 E31F BNE ERSS05 YES, ERASE COLOR IS CHAR'S COLOR.
00755A E31D 36 2F A LDB ..BCL
00756 E31F A ERSS05 EQU #
00757A E321 8D 47 E368 BSR ERSV BLUE
00758A E323 8D 45 E368 BSR ERSV RED
00759A E325 20 43 E368 BSA ERSV GREEN & RETURN
00760 E327 A
00761 E329 A ERSS0F EQU #
00762A E32B SCROFF
00763A E32D 00 23 A ERSSUB STA CRTON
00764A E32A 8D A7 E2D3 BSR ERSS0F
00765A E32C 00 23 A NEG CCLCDS
00766A E32E SCRON
00767A E32E 65 D408 A BITA CRTON

```

```

00767A E331 96 7E A ERSBUF LDA
00768A E333 BD E208 A JSR
00769A E335 34 20 A A PSHS
00770A E338 33 84 A LEAU
00771A E33A 96 70 A LDA
00772A E33C BD E208 A JSR
00773A E33F 34 20 A A PSHS
00774A E341 4F 04 E348 A CLRA
00775A E342 8D 04 E348 BSR
00776A E344 96 2E A LDA
00777A E346 35 50 A PULS
00778 E348 A
00779 E348 A ERSARA EQU #
00780 E348 A
00781 E348 A
00782 E348 A # A STORED DATA
00783 E348 A # X START ADDR
00784 E348 A # U END ADDR
00785 E348 A
00786 E348 A # THESE PARAMETERS
00787 E348 A
00788 E348 A # THE X IS LESS
00789 E348 A # SIZE (U-X) IS
00790 E348 A # GREATER THAN 5.
00791 E348 A
00792A E348 31 02 A LEAY
00793A E34A 30 06 A LEAX
00794A E34C 34 89 A PSHS
00795A E34E 1F 89 A TFR
00796A E350 1F 01 A TFR
00797A E352 31 84 A LEAY
00798A E354 36 36 A ERSAR3 PSNU
00799A E356 11A3 E4 A CMPLU
00800A E359 24 F9 E354 A BHS
00801A E35B 11A3 62 A CMPLU
00802A E35E 25 05 E366 A BLD
00803A E360 27 02 E364 A BEQ
00804A E362 ED C3 A STD
00805A E364 ED C3 A ERSAR8 STD
00806A E366 A6 A ERSAR3 RTS
00807 E368 A
00808A E368 44 52 A ERSV LSRA
00809A E369 34 00 A PSHS
00810A E36B 86 00 A LDA
00811A E36D 24 01 E370 A BCC
00812A E36F 43 COMA
00813A E370 ERSV05 VACON
00814A E370 B5 D408 A BITA
00815A E373 8D 03 E348 BSR
00816A E375 67 D409 A VACOFF
00817A E377 35 52 A STA
00818A E37A 30 89 4000 A PULS
00819A E37E 33 C9 4000 A LEAX
00820 E382 39 C9 4000 A LEAU
00821 E384 39 C9 4000 A RTS
00822 E386 A
00823 E388 A
00824 E38A A
00825A E38B 9E E383 A GAXC EQU
00826A E385 109E 64 A LDX
00827 E387 62 A LOY
00828A E388 8C E388 A GAXCP EQU
00829A E38B 24 04 E391 A GAXC1 CMPA
00830A E38D 108C 00C8 A BHS
00831A E391 39 00C8 A CMPLU
00832 E393 A
00833 E395 A
00834A E39C 80 A BAS FCB
00835A E39C 40 A FCB
00836A E39A 20 A FCB
00837A E39B 10 A FCB
00838A E39C 08 A FCB
00839A E397 04 A FCB
00840A E398 02 A FCB
00841A E399 01 A FCB
00842 E39A A
00843 E39B A
00844 E39C A

```

```

00845 E39D A
00846 E39E A
00847 E39F A # GPGETR I READ
00848 E3A0 A
00849 E3A1 A
00849A E39A E39A A GPGETR EQU
00850A E39A A GPGET1 VACON
00851A E39D 4F D409 A CLRA
00852A E39E E5 C4 A BITB
00853A E3A0 27 01 E3A3 A BEQ
00854A E3A2 4C 01 A INCA
00855A E3A3 E5 C9 4000 A GPGET2 BITB
00856A E3A7 27 02 E3AB A EQ
00857A E3A9 8B 02 A ADDA
00858A E3AB E5 C9 8000 A GPGET3 BITB
00859A E3AF 27 02 E3B3 A BEQ
00860A E3B1 8B 04 A ADDA
00861A E3B3 A GPGET4 VACOFF
00862A E3B7 D409 A STA
00863A E3B6 39 A RTS
00864 E3B8 A
00865 E3B9 A
00866 E3BA A
00867 E3BB A
00868 E3BC A
00869 E3BD A
00870 E3BE A
00871A E3B7 34 E3B7 A CFADR EQU
00872A E3B8 C6 08 A PSIS
00873A E3B8 3D 08 A LDB
00874A E3BC C3 D800 A MUL
00875A E3BF 1F 01 A ADDD
00876A E3C1 35 86 A TFR
00877 E3C1 A RTS
00878 E3C2 A
00879 E3C3 A
00880 E3C4 A
00881 E3C5 A # CFADR I CALCULATE

```

.CVTOP+1 GET BUFFER ADDR THAT IS  
BADR1 PAST 1 TO END OF ERASE AREA.  
Y SAVE ATTR ADDR.  
.X SET U CHAR ADDR  
.CVTOP GET BUFFER ADDRESS THAT  
BADR1 IS TOP OF ERASE AREA.  
Y SAVE ATTR ADDR.

ERSARA ERASE CHAR BUFFER.

..ACL  
X,U ERASE ATTR BUFFER.

\*

MUST BE AS FOLLOWS.

THAN THE U.  
EVEN NUMBER AND

2,X  
6,X  
X,Y  
A,B MAKE DATA  
D,X  
.X  
D,X,Y SET 6 BYTES.  
.S IS THE REMAINDER .GE.6?  
ERSAR3 YES.  
2,S HOW MANY BYTES IS REMAIND?  
ERSAR3 IT IS 0.  
ERSAF8 IT IS 2.  
..U IT IS 4.  
..U  
(D,Y)  
D,Y,PC

A,X,U  
#S00  
ERSV05  
  
VACC  
ERSARA  
  
VACC  
A,X,U  
\$4000,X TOP ADDRESS OF NEXT COLOR.  
\$4000,U END ADDRESS OF NEXT COLOR.

COMMON SUBROUTINE  
GRAPHIC CO-ORDINATE X,Y

.\*  
.GX  
.GY  
.\*  
#GX0  
GAXC2  
#GYD

\$S0  
\$40  
\$20  
\$10  
\$08  
\$04  
\$02  
\$01

COLOR CODE

.\*

VACC

..VB,U  
GPGET2

..VR,U  
GPGET3

#2

..VG,U  
GPGET4

#4

VACC

CHARACTER FONT ADDRESS

.\*

A,B

#CFH

..CFNT

D,X

(A,B)

A,B,PC

BUFFER ADDRESS

```

00882      E3C3      A  CRADR EQU      #
00883      58        A  LDD      .CY
00884A E3C3 DC      A  JSR      BADR
00885A E3C5 BD      A  STX      .CBAD
00886A E3C8 3F      A  STY      .CAAD
00887A E3CA 103F    A  LDD      .CY
00888A E3CD DC      A  JSR      BVAD
00889A E3CF BD      A  STD      .CVAD
00890A E3D2 DD      A  RTS
00891A E3D4 39      #
00892      #
00893      #
00894      #
00895      #
00896      #
00897      #
00898      #
00899      #
00900A E3D5 31      A  CAXCR EQU      #
00901A E3D7 24      E3D5  CMPA   .LNC
00902A E3D9 01      E3D5  BCC    CAXC2
00903A E3DB 39      A  CMPB   .DMAP+2+3
00904      CAXC2    RTS
00905      #
00906      #
00907A E3DC 34      A  GADDF EQU      #
00908A E3DE EC      A  PSHS   A,X,Y
00909A E3E0 8D      A  LDD      1,S
00910A E3E2 1E      E3F4  BSR    GXOFFF
00911A E3E4 85      A  EXG    D,X
00912A E3E6 3D      A  LDA    #GXSZ
00913A E3E7 33      A  MUL
00914A E3E9 CC      A  LEAU   D,X
00915A E3EB 1E      A  LDD    #BAS
00916A E3ED E5      A  EXG    D,Y
00917A E3F0 35      A  LDB    B,Y
00918      A 3F0 35  B2  A  RTSPUL (A,X,Y)
00919      PULS   A,X,Y,PC
00920      #
00921      #
00922      #
00923      #
00924      #
00925A E3F2 1F      A  GXOFF EQU      #
00926A E3F4 8D      E3F2  TFP    D,X
00927A E3F6 1E      E3F6  GXOFFF BSR    LSRD3
00928A E3F8 C4      A  EXG    D,X
00929A E3FA 39      A  ANDB   #07
00930      RTS
00931      #
00932      #
00933      #
00934A E3FE 44      E3FE  A  LSRD3 EQU      #
00935A E3FC 55      A  LSPA   LSPA
00936A E3FD 44      A  POPB   LSPA
00937A E3FE 55      A  POPB   LSPA
00938A E3FF 44      A  LSPA   LSPA
00939A E400 55      A  POPB   LSPA
00940A E401 39      A  RTS
00941      #
00942A E402 SE      D383  A  BOXINV LDX    #CMOPRM
00943A E404 13E      D383  A  LDY    #,BOX1
00944A E405 C6      A  LDB    #
00945A E406 8D      E44A  BSR    MOVE
00946A E408 8D      E41A  BSR    BOX
00947A E40F DC      A  LDD    .BOXXC
00948A E411 38      A  SUBD   .BOXX1
00949A E413 3F      A  TFR    D,X
00950A E415 DC      A  LDD    .BOXY2
00951A E417 38      A  SUBD   .BOXY1
00952A E419 39      A  RTS
00953      #
00954      #
00955      #
00956      #
00957      #
00958      #
00959A E41A CE      E41A  A  BOX EQU      #
00960A E41D AE      D383  A  LDU    #,BOX1 TOP ADDRESS U
00961      A  BOXF   LDX    ,U

```

```

00961A E41F 10AE 42      A  LDY    2,U
00962A E422 8D      E442  BSR    GAXCH
00963A E424 AE      A  LDX    4,U
00964A E426 10AE 45      A  LDY    6,U
00965A E428 8D      E442  BSR    GAXCH
00966A E42E AC      A  CMPX   ,U
00967A E430 24      E435  BHS    BOX1
00968A E432 EC      A  LDD    ,U
00969A E434 AF      A  STX    ,U
00970A E436 ED      A  STD    4,U
00971A E438 10AC 42      A  BOX1  CMPY   2,U
00972A E43B 24      E441  BHS    BOX2
00973A E43A EC      A  LDD    2,U
00974A E43C 10AF 42      A  STY    2,U
00975A E43F ED      A  STD    6,U
00976A E441 39      A  BOX2  RTS
00977      #
00978A E442 BD      E388  A  GAXCH JSR    GAXCF
00979A E445 25      FA  E441  BCS    BOX2
00980A E447 7E      E0D8  A  JMP    ERR63
00981      #
00982      #
00983      #
00984      #
00985      #
00986      #
00987      #
00988      #
00989      #
00990A E44A A6      E44A  A  MOVE EQU      #
00991A E44C A7      A  LDA    ,X+
00992A E44E 5A      A  STA    ,Y+
00993A E44F 26      F9  E44A  BNE    MOVE
00994A E451 39      A  RTS
00995      #
00996      #
00997      #
00998      #
00999      #
01000      #
01001      #

```

```

00995A E41F 10AE 42      A  LDY    2,U
00996A E422 8D      E442  BSR    GAXCH
00997A E424 AE      A  LDX    4,U
00998A E426 10AE 45      A  LDY    6,U
00999A E428 8D      E442  BSR    GAXCH
01000A E42E AC      A  CMPX   ,U
01001A E430 24      E435  BHS    BOX1
01002A E432 EC      A  LDD    ,U
01003A E434 AF      A  STX    ,U
01004A E436 ED      A  STD    4,U
01005A E438 10AC 42      A  BOX1  CMPY   2,U
01006A E43B 24      E441  BHS    BOX2
01007A E43A EC      A  LDD    2,U
01008A E43C 10AF 42      A  STY    2,U
01009A E43F ED      A  STD    6,U
01010A E441 39      A  BOX2  RTS
01011      #
01012A E442 BD      E388  A  GAXCH JSR    GAXCF
01013A E445 25      FA  E441  BCS    BOX2
01014A E447 7E      E0D8  A  JMP    ERR63
01015      #
01016      #
01017      #
01018      #
01019      #
01020      #
01021      #

```

```

01021A E442 BD      E388  A  GAXCH JSR    GAXCF
01022A E445 25      FA  E441  BCS    BOX2
01023A E447 7E      E0D8  A  JMP    ERR63
01024      #
01025      #
01026      #
01027      #
01028      #
01029      #
01030      #
01031      #
01032      #
01033      #
01034      #
01035      #
01036      #
01037      #
01038      #
01039      #
01040      #
01041      #
01042      #
01043      #
01044      #
01045      #
01046      #
01047      #
01048      #
01049      #
01050      #
01051      #
01052      #
01053      #
01054      #
01055      #
01056      #
01057      #
01058      #
01059      #
01060      #
01061      #
01062      #
01063      #
01064      #
01065      #
01066      #
01067      #
01068      #
01069      #
01070      #
01071      #
01072      #
01073      #
01074      #
01075      #
01076      #
01077      #
01078      #
01079      #
01080      #
01081      #
01082      #
01083      #
01084      #
01085      #
01086      #
01087      #
01088      #
01089      #
01090      #
01091      #
01092      #
01093      #
01094      #
01095      #
01096      #
01097      #
01098      #
01099      #
01100      #
01101      #

```

# FM-7 サブシステム・モニタ ソースリスト

```

01002A E452 7F D0A0 A INITLN CLF LNCHR
01003A E455 DC 60 A LDD .GCL LOAD COLOR AND FUNCTION CODE.
01004A E457 8E F0B0 A LDX #GDFUNC LOAD FUNCTION ENTRY TABLE ADDRESS.
01005A E45A CE D0B0 A LDU #BLUFIN LOAD RAM HOOK TABLE ADDRESS.
01006A E45D 8D 12 E471 BSR LENTRY SET THE RAM HOOKS.
01007A E45F DC 60 A LDD .GCL SEP. 3 1981
01008A E461 8E F07E A LDX #LWFWNC PARALLEL TO X AXIS LINE'S SIDE TO SET ENTRY
01009A E464 CE D079 A LDU #BLUUSD
01010A E467 8D 08 E471 BSR LENTRY
                                AND DO GPDSP INITIALIZE.
01011
01012
01013
01014
01015
01016 E469 DC E469 A GPDSPI EQU #
01017A E46B 8E E4AD A GPDSIF LDX #GDFUNC MAKE RAM HOOK FOR DOT DISPLAY.
01018A E46E CE D0B3 A FENTRY LDU #BLUUSD LOAD RAM HOOK ADDRESS.
01019A E471 C1 01 E477 BSR LENTRY CMPB NOTPRE NO,
01020A E473 76 02 E477 BSR LENTRY LDB .NOTPRE YES, USE BACK COLOR.
01021A E475 36 2F A NOTPRE LSLB
01022A E477 58 A NOTPRE LSLB
01023A E479 3A A NOTPRE AEX
01024A E47B 58 A NOTPRE LDB #7E
01025A E47D 3A A NOTPRE BSR SETENT
01026A E47E C6 7E A NOTPRE LDB #7E
01027A E47F 8D 02 E480 BSR SETENT
01028A E47E 8D 00 E480 BSR SETENT
01029A E480 E7 C0 A SETENT STB .U+
01030A E482 44 A SETENT LSRB
01031A E483 34 06 A SETENT PSMS
01032A E485 CE 34 A SETENT LDD .X
01033A E487 25 02 E48E BCS CEITON
01034A E489 EC 02 A LDD .X
01035A E48B ED C1 A CEITON STB .U++
01036A E48D 80 A CEITON RTS
01037A E48D 80 A CEITON PULS
01038
01039
01040
                                GPDSP : DISPLAY DOT ON GRAPHIC CO-ORDINATE

```

```

01041
01042
01043 E48F 9E E48F A GPSP EQU #
01044A E491 9E 64 A GPSP LDX .GX LOAD X COORDINATES
01045A E493 10 6E 6C A GPSP LDY .GY LOAD Y COORDINATES
01046A E494 ED 82 A GPSP JSP GADDP VRAM ADDR. MADE
01047A E497 34 04 A GPSP PSMS
01048A E499 30 33 A GPSP JSP ELUDSP
01049A E49B E6 04 A GPSP LEAU .VR,U
01050A E49D 33 03 4000 A GPSP JSP REODSP
01051A E4A1 30 36 A GPSP LDB .X
01052A E4A3 E6 04 A GPSP LEAU .VR,U
01053A E4A5 33 03 4000 A GPSP JSP REODSP
01054A E4A7 30 38 A GPSP JSP REODSP
01055A E4A9 30 38 A GPSP JSP REODSP
01056A E4AB 35 34 A GPSP RTS
01057A E4AD 40 A GPSP FDB GPOFA,GPAND SE1
01058A E4AF 40 A GPSP FDB GPOFA,GPAND PRESET
01059A E4B1 40 A GPSP FDB GPOFA,GPAND OF
01060A E4B3 40 A GPSP FDB GPOFA,GPAND OF
01061A E4B5 40 A GPSP FDB GPOFA,GPAND OF
01062A E4B7 40 A GPSP FDB GPOFA,GPAND OF
01063A E4B9 40 A GPSP FDB GPOFA,GPAND OF
01064A E4BB 40 A GPSP FDB GPOFA,GPAND OF
01065A E4BD 40 A GPSP FDB GPOFA,GPAND OF
01066A E4BF 40 A GPSP FDB GPOFA,GPAND OF
01067A E4C1 40 A GPSP FDB GPOFA,GPAND OF
01068A E4C3 40 A GPSP FDB GPOFA,GPAND OF
01069A E4C5 40 A GPSP FDB GPOFA,GPAND OF
01070A E4C7 40 A GPSP FDB GPOFA,GPAND OF
01071A E4C9 40 A GPSP FDB GPOFA,GPAND OF
01072A E4CB 40 A GPSP FDB GPOFA,GPAND OF
01073A E4CD 40 A GPSP FDB GPOFA,GPAND OF
01074A E4CE 40 A GPSP FDB GPOFA,GPAND OF
01075A E4CF 40 A GPSP FDB GPOFA,GPAND OF
01076A E4D1 40 A GPSP FDB GPOFA,GPAND OF
01077A E4D3 40 A GPSP FDB GPOFA,GPAND OF
01078A E4D5 40 A GPSP FDB GPOFA,GPAND OF
01079A E4D7 40 A GPSP FDB GPOFA,GPAND OF
01080A E4D9 40 A GPSP FDB GPOFA,GPAND OF
01081A E4DB 40 A GPSP FDB GPOFA,GPAND OF
01082A E4DD 40 A GPSP FDB GPOFA,GPAND OF
01083A E4DE 40 A GPSP FDB GPOFA,GPAND OF
01084A E4DF 40 A GPSP FDB GPOFA,GPAND OF
01085A E4E1 40 A GPSP FDB GPOFA,GPAND OF
01086A E4E3 40 A GPSP FDB GPOFA,GPAND OF
01087A E4E5 40 A GPSP FDB GPOFA,GPAND OF
01088A E4E7 40 A GPSP FDB GPOFA,GPAND OF
01089A E4E9 40 A GPSP FDB GPOFA,GPAND OF
01090A E4EB 40 A GPSP FDB GPOFA,GPAND OF
01091A E4ED 40 A GPSP FDB GPOFA,GPAND OF
01092A E4EF 40 A GPSP FDB GPOFA,GPAND OF
01093A E4F1 40 A GPSP FDB GPOFA,GPAND OF
01094A E4F3 40 A GPSP FDB GPOFA,GPAND OF
01095A E4F5 40 A GPSP FDB GPOFA,GPAND OF
01096A E4F7 40 A GPSP FDB GPOFA,GPAND OF
01097A E4F9 40 A GPSP FDB GPOFA,GPAND OF
01098A E4FB 40 A GPSP FDB GPOFA,GPAND OF
01099A E4FD 40 A GPSP FDB GPOFA,GPAND OF
01100A E4FF 40 A GPSP FDB GPOFA,GPAND OF
01101A E501 40 A GPSP FDB GPOFA,GPAND OF
01102A E503 40 A GPSP FDB GPOFA,GPAND OF
01103A E505 40 A GPSP FDB GPOFA,GPAND OF
01104A E507 40 A GPSP FDB GPOFA,GPAND OF
01105A E509 40 A GPSP FDB GPOFA,GPAND OF
01106A E50B 40 A GPSP FDB GPOFA,GPAND OF
01107A E50D 40 A GPSP FDB GPOFA,GPAND OF
01108A E50F 40 A GPSP FDB GPOFA,GPAND OF
01109A E511 40 A GPSP FDB GPOFA,GPAND OF
01110A E513 40 A GPSP FDB GPOFA,GPAND OF
01111A E515 40 A GPSP FDB GPOFA,GPAND OF
01112A E517 40 A GPSP FDB GPOFA,GPAND OF
01113A E519 40 A GPSP FDB GPOFA,GPAND OF
01114A E51B 40 A GPSP FDB GPOFA,GPAND OF
01115A E51D 40 A GPSP FDB GPOFA,GPAND OF
01116A E51F 40 A GPSP FDB GPOFA,GPAND OF
01117A E521 40 A GPSP FDB GPOFA,GPAND OF
01118A E523 40 A GPSP FDB GPOFA,GPAND OF
01119A E525 40 A GPSP FDB GPOFA,GPAND OF
01120A E527 40 A GPSP FDB GPOFA,GPAND OF
01121A E529 40 A GPSP FDB GPOFA,GPAND OF
01122A E52B 40 A GPSP FDB GPOFA,GPAND OF
01123A E52D 40 A GPSP FDB GPOFA,GPAND OF
01124A E52F 40 A GPSP FDB GPOFA,GPAND OF
01125A E531 40 A GPSP FDB GPOFA,GPAND OF
01126A E533 40 A GPSP FDB GPOFA,GPAND OF
01127A E535 40 A GPSP FDB GPOFA,GPAND OF
01128A E537 40 A GPSP FDB GPOFA,GPAND OF
01129A E539 40 A GPSP FDB GPOFA,GPAND OF
01130A E53B 40 A GPSP FDB GPOFA,GPAND OF
01131A E53D 40 A GPSP FDB GPOFA,GPAND OF
01132A E53F 40 A GPSP FDB GPOFA,GPAND OF
01133A E541 40 A GPSP FDB GPOFA,GPAND OF
01134A E543 40 A GPSP FDB GPOFA,GPAND OF
01135A E545 40 A GPSP FDB GPOFA,GPAND OF
01136A E547 40 A GPSP FDB GPOFA,GPAND OF
01137A E549 40 A GPSP FDB GPOFA,GPAND OF
01138A E54B 40 A GPSP FDB GPOFA,GPAND OF
01139A E54D 40 A GPSP FDB GPOFA,GPAND OF
01140A E54F 40 A GPSP FDB GPOFA,GPAND OF
01141A E551 40 A GPSP FDB GPOFA,GPAND OF
01142A E553 40 A GPSP FDB GPOFA,GPAND OF
01143A E555 40 A GPSP FDB GPOFA,GPAND OF
01144A E557 40 A GPSP FDB GPOFA,GPAND OF
01145A E559 40 A GPSP FDB GPOFA,GPAND OF
01146A E55B 40 A GPSP FDB GPOFA,GPAND OF
01147A E55D 40 A GPSP FDB GPOFA,GPAND OF
01148A E55F 40 A GPSP FDB GPOFA,GPAND OF
01149A E561 40 A GPSP FDB GPOFA,GPAND OF
01150A E563 40 A GPSP FDB GPOFA,GPAND OF
01151A E565 40 A GPSP FDB GPOFA,GPAND OF
01152A E567 40 A GPSP FDB GPOFA,GPAND OF
01153A E569 40 A GPSP FDB GPOFA,GPAND OF
01154A E56B 40 A GPSP FDB GPOFA,GPAND OF
01155A E56D 40 A GPSP FDB GPOFA,GPAND OF
01156A E56F 40 A GPSP FDB GPOFA,GPAND OF
01157A E571 40 A GPSP FDB GPOFA,GPAND OF
01158A E573 40 A GPSP FDB GPOFA,GPAND OF
01159A E575 40 A GPSP FDB GPOFA,GPAND OF
01160A E577 40 A GPSP FDB GPOFA,GPAND OF
01161A E579 40 A GPSP FDB GPOFA,GPAND OF
01162A E57B 40 A GPSP FDB GPOFA,GPAND OF
01163A E57D 40 A GPSP FDB GPOFA,GPAND OF
01164A E57F 40 A GPSP FDB GPOFA,GPAND OF
01165A E581 40 A GPSP FDB GPOFA,GPAND OF
01166A E583 40 A GPSP FDB GPOFA,GPAND OF
01167A E585 40 A GPSP FDB GPOFA,GPAND OF
01168A E587 40 A GPSP FDB GPOFA,GPAND OF
01169A E589 40 A GPSP FDB GPOFA,GPAND OF
01170A E58B 40 A GPSP FDB GPOFA,GPAND OF
01171A E58D 40 A GPSP FDB GPOFA,GPAND OF
01172A E58F 40 A GPSP FDB GPOFA,GPAND OF
01173A E591 40 A GPSP FDB GPOFA,GPAND OF
01174A E593 40 A GPSP FDB GPOFA,GPAND OF
01175A E595 40 A GPSP FDB GPOFA,GPAND OF
01176A E597 40 A GPSP FDB GPOFA,GPAND OF
01177A E599 40 A GPSP FDB GPOFA,GPAND OF
01178A E59B 40 A GPSP FDB GPOFA,GPAND OF
01179A E59D 40 A GPSP FDB GPOFA,GPAND OF
01180A E59F 40 A GPSP FDB GPOFA,GPAND OF
01181A E5A1 40 A GPSP FDB GPOFA,GPAND OF
01182A E5A3 40 A GPSP FDB GPOFA,GPAND OF
01183A E5A5 40 A GPSP FDB GPOFA,GPAND OF
01184A E5A7 40 A GPSP FDB GPOFA,GPAND OF
01185A E5A9 40 A GPSP FDB GPOFA,GPAND OF
01186A E5AB 40 A GPSP FDB GPOFA,GPAND OF
01187A E5AD 40 A GPSP FDB GPOFA,GPAND OF
01188A E5AF 40 A GPSP FDB GPOFA,GPAND OF
01189A E5B1 40 A GPSP FDB GPOFA,GPAND OF
01190A E5B3 40 A GPSP FDB GPOFA,GPAND OF
01191A E5B5 40 A GPSP FDB GPOFA,GPAND OF
01192A E5B7 40 A GPSP FDB GPOFA,GPAND OF
01193A E5B9 40 A GPSP FDB GPOFA,GPAND OF
01194A E5BB 40 A GPSP FDB GPOFA,GPAND OF
01195A E5BD 40 A GPSP FDB GPOFA,GPAND OF
01196A E5BF 40 A GPSP FDB GPOFA,GPAND OF
01197A E5C1 40 A GPSP FDB GPOFA,GPAND OF
01198A E5C3 40 A GPSP FDB GPOFA,GPAND OF
01199A E5C5 40 A GPSP FDB GPOFA,GPAND OF
01200A E5C7 40 A GPSP FDB GPOFA,GPAND OF
01201A E5C9 40 A GPSP FDB GPOFA,GPAND OF
01202A E5CB 40 A GPSP FDB GPOFA,GPAND OF
01203A E5CD 40 A GPSP FDB GPOFA,GPAND OF
01204A E5CE 40 A GPSP FDB GPOFA,GPAND OF
01205A E5CF 40 A GPSP FDB GPOFA,GPAND OF
01206A E5D1 40 A GPSP FDB GPOFA,GPAND OF
01207A E5D3 40 A GPSP FDB GPOFA,GPAND OF
01208A E5D5 40 A GPSP FDB GPOFA,GPAND OF
01209A E5D7 40 A GPSP FDB GPOFA,GPAND OF
01210A E5D9 40 A GPSP FDB GPOFA,GPAND OF
01211A E5DB 40 A GPSP FDB GPOFA,GPAND OF
01212A E5DD 40 A GPSP FDB GPOFA,GPAND OF
01213A E5DE 40 A GPSP FDB GPOFA,GPAND OF
01214A E5DF 40 A GPSP FDB GPOFA,GPAND OF
01215A E5E1 40 A GPSP FDB GPOFA,GPAND OF
01216A E5E3 40 A GPSP FDB GPOFA,GPAND OF
01217A E5E5 40 A GPSP FDB GPOFA,GPAND OF
01218A E5E7 40 A GPSP FDB GPOFA,GPAND OF
01219A E5E9 40 A GPSP FDB GPOFA,GPAND OF
01220A E5EB 40 A GPSP FDB GPOFA,GPAND OF
01221A E5ED 40 A GPSP FDB GPOFA,GPAND OF
01222A E5EF 40 A GPSP FDB GPOFA,GPAND OF
01223A E5F1 40 A GPSP FDB GPOFA,GPAND OF
01224A E5F3 40 A GPSP FDB GPOFA,GPAND OF
01225A E5F5 40 A GPSP FDB GPOFA,GPAND OF
01226A E5F7 40 A GPSP FDB GPOFA,GPAND OF
01227A E5F9 40 A GPSP FDB GPOFA,GPAND OF
01228A E5FB 40 A GPSP FDB GPOFA,GPAND OF
01229A E5FD 40 A GPSP FDB GPOFA,GPAND OF
01230A E5FF 40 A GPSP FDB GPOFA,GPAND OF

```

```

01073 E4CD 53 E4CD A GPAND EQU #
01074A E4CE 53 E4CD A GPAND EQU #
01075A E4CF 53 E4CD A GPAND EQU #
01076A E4D1 12 D409 A GPAND EQU #
01077A E4D3 12 D409 A GPAND EQU #
01078A E4D5 12 D409 A GPAND EQU #
01079A E4D7 12 D409 A GPAND EQU #
01080A E4D9 12 D409 A GPAND EQU #
01081A E4DB 12 D409 A GPAND EQU #
01082A E4DD 12 D409 A GPAND EQU #
01083A E4DE 12 D409 A GPAND EQU #
01084A E4DF 12 D409 A GPAND EQU #
01085A E4E1 12 D409 A GPAND EQU #
01086A E4E3 12 D409 A GPAND EQU #
01087A E4E5 12 D409 A GPAND EQU #
01088A E4E7 12 D409 A GPAND EQU #
01089A E4E9 12 D409 A GPAND EQU #
01090A E4EB 12 D409 A GPAND EQU #
01091A E4ED 12 D409 A GPAND EQU #
01092A E4EF 12 D409 A GPAND EQU #
01093A E4F1 12 D409 A GPAND EQU #
01094A E4F3 12 D409 A GPAND EQU #
01095A E4F5 12 D409 A GPAND EQU #
01096A E4F7 12 D409 A GPAND EQU #
01097A E4F9 12 D409 A GPAND EQU #
01098A E4FB 12 D409 A GPAND EQU #
01099A E4FD 12 D409 A GPAND EQU #
01100A E4FF 12 D409 A GPAND EQU #
01101A E501 12 D409 A GPAND EQU #
01102A E503 12 D409 A GPAND EQU #
01103A E505 12 D409 A GPAND EQU #
01104A E507 12 D409 A GPAND EQU #
01105A E509 12 D409 A GPAND EQU #
01106A E50B 12 D409 A GPAND EQU #
01107A E50D 12 D409 A GPAND EQU #
01108A E50F 12 D409 A GPAND EQU #
01109A E511 12 D409 A GPAND EQU #
01110A E513 12 D409 A GPAND EQU #
01111A E515 12 D409 A GPAND EQU #
01112A E517 12 D409 A GPAND EQU #
01113A E519 12 D409 A GPAND EQU #
01114A E51B 12 D409 A GPAND EQU #
01115A E51D 12 D409 A GPAND EQU #
01116A E51F 12 D409 A GPAND EQU #
01117A E521 12 D409 A GPAND EQU #
01118A E523 12 D409 A GPAND EQU #
01119A E525 12 D409 A GPAND EQU #
01120A E527 12 D409 A GPAND EQU #
01121A E529 12 D409 A GPAND EQU #
01122A E52B 12 D409 A GPAND EQU #
01123A E52D 12 D409 A GPAND EQU #
01124A E52F 12 D409 A GPAND EQU #
01125A E531 12 D409 A GPAND EQU #
01126A E533 12 D409 A GPAND EQU #
01127A E535 12 D409 A GPAND EQU #
01128A E537 12 D409 A GPAND EQU #
01129A E539 12 D409 A GPAND EQU #
01130A E53B 12 D409 A GPAND EQU #
01131A E53D 12 D409 A GPAND EQU #
01132A E53F 12 D409 A GPAND EQU #
01133A E541 12 D409 A GPAND EQU #
01134A E543 12 D409 A GPAND EQU #
01135A E545 12 D409 A GPAND EQU #
01136A E547 12 D409 A GPAND EQU #
01137A E549 12 D409 A GPAND EQU #
01138A E54B 12 D409 A GPAND EQU #
01139A E54D 12 D409 A GPAND EQU #
01140A E54F 12 D409 A GPAND EQU #
01141A E551 12 D409 A GPAND EQU #
01142A E553 12 D409 A GPAND EQU #
01143A E555 12 D409 A GPAND EQU #
01144A E557 12 D409 A GPAND EQU #
01145A E559 12 D409 A GPAND EQU #
01146A E55B 12 D409 A GPAND EQU #
01147A E55D 12 D409 A GPAND EQU #
01148A E55F 12 D409 A GPAND EQU #
01149A E561 12 D409 A GPAND EQU #
01150A E563 12 D409 A GPAND EQU #
01151A E565 12 D409 A GPAND EQU #
01152A E567 12 D409 A GPAND EQU #
01153A E569 12 D409 A GPAND EQU #
01154A E56B 12 D409 A GPAND EQU #
01155A E56D 12 D409 A GPAND EQU #
01156A E56F 12 D409 A GPAND EQU #
01157A E571 12 D409 A GPAND EQU #
01158A E573 12 D409 A GPAND EQU #
01159A E575 12 D409 A GPAND EQU #
01160A E577 12 D409 A GPAND EQU #
01161A E579 12 D409 A GPAND EQU #
01162A E57B 12 D409 A GPAND EQU #
01163A E57D 12 D409 A GPAND EQU #
01164A E57F 12 D409 A GPAND EQU #
01165A E581 12 D409 A GPAND EQU #
01166A E583 12 D409 A GPAND EQU #
01167A E585 12 D409 A GPAND EQU #
01168A E587 12 D409 A GPAND EQU #
01169A E589 12 D409 A GPAND EQU #
01170A E58B 12 D409 A GPAND EQU #
01171A E58D 12 D409 A GPAND EQU #
01172A E58F 12 D409 A GPAND EQU #
01173A E591 12 D409 A GPAND EQU #
01174A E593 12 D409 A GPAND EQU #
01175A E595 12 D409 A GPAND EQU #
01176A E597 12 D409 A GPAND EQU #
01177A E599 12 D409 A GPAND EQU #
01178A E59B 12 D409 A GPAND EQU #
01179A E59D 12 D409 A GPAND EQU #
01180A E59F 12 D409 A GPAND EQU #
01181A E5A1 12 D409 A GPAND EQU #
01182A E5A3 12 D409 A GPAND EQU #
01183A E5A5 12 D409 A GPAND EQU #
01184A E5A7 12 D409 A GPAND EQU #
01185A E5A9 12 D409 A GPAND EQU #
01186A E5AB 12 D409 A GPAND EQU #
01187A E5AD 12 D409 A GPAND EQU #
01188A E5AF 12 D409 A GPAND EQU #
01189A E5B1 12 D409 A GPAND EQU #
01190A E5B3 12 D409 A GPAND EQU #
01191A E5B5 12 D409 A GPAND EQU #
01192A E5B7 12 D409 A GPAND EQU #
01193A E5B9 12 D409 A GPAND EQU #
01194A E5BB 12 D409 A GPAND EQU #
01195A E5BD 12 D409 A GPAND EQU #
01196A E5BF 12 D409 A GPAND EQU #
01197A E5C1 12 D409 A GPAND EQU #
01198A E5C3 12 D409 A GPAND EQU #
01199A E5C5 12 D409 A GPAND EQU #
01200A E5C7 12 D409 A GPAND EQU #
01201A E5C9 12 D409 A GPAND EQU #
01202A E5CB 12 D409 A GPAND EQU #
01203A E5CD 12 D409 A GPAND EQU #
01204A E5CE 12 D409 A GPAND EQU #
01205A E5CF 12 D409 A GPAND EQU #
01206A E5D1 12 D409 A GPAND EQU #
01207A E5D3 12 D409 A GPAND EQU #
01208A E5D5 12 D409 A GPAND EQU #
01209A E5D7 12 D409 A GPAND EQU #
01210A E5D9 12 D409 A GPAND EQU #
01211A E5DB 12 D409 A GPAND EQU #
01212A E5DD 12 D409 A GPAND EQU #
01213A E5DE 12 D409 A GPAND EQU #
01214A E5DF 12 D409 A GPAND EQU #
01215A E5E1 12 D409 A GPAND EQU #
01216A E5E3 12 D409 A GPAND EQU #
01217A E5E5 12 D409 A GPAND EQU #
01218A E5E7 12 D409 A GPAND EQU #
01219A E5E9 12 D409 A GPAND EQU #
01220A E5EB 12 D409 A GPAND EQU #
01221A E5ED 12 D409 A GPAND EQU #
01222A E5EF 12 D409 A GPAND EQU #
01223A E5F1 12 D409 A GPAND EQU #
01224A E5F3 12 D409 A GPAND EQU #
01225A E5F5 12 D409 A GPAND EQU #
01226A E5F7 12 D409 A GPAND EQU #
01227A E5F9 12 D409 A GPAND EQU #
01228A E5FB 12 D409 A GPAND EQU #
01229A E5FD 12 D409 A GPAND EQU #
01230A E5FF 12 D409 A GPAND EQU #

```

```

01110A E4FA 5F CLRB .S
01111A E4FB 64 LSP
01112A E4FD 59 RLB
01113A E4FE 64 LSP
01114A E500 59 RLB
01115A E501 59 RLB
01116A E502 EE 85 A LDU B,X
01117A E504 EF A1 A STU Y++
01118A E506 4A DECA
01119A E507 26 BNE
01120A E509 36 ED E4F6 RTSPUL
                                (D,X,PC)
                                PULS
01121
01122
01123A E50B 44 E50B A BITEXT EQU #
01124A E50C 56 LSRB
01125A E50D 57 RLB
01126A E50E 44 ASRB
01127A E50F 56 ASRB
01128A E510 57 ASRB
01129A E511 44 LSRB
01130A E512 56 RLB
01131A E513 57 ASRB
01132A E514 44 LSRB
01133A E515 56 RLB
01134A E516 57 ASRB
01135A E517 39 RTS
01136
01137A E518 9E 5E A CDSPC LDX .C
01138A E51A 8C A CDSPC SKIP2
01139A E51B 9E 4E A CDSPC FCB
01140A E51D 33 E51D A CDSPC EQU
01141A E51D 33 E51D A CDSPC LEAU
01142A E51F 00 E3E7 A JSP
01143A E520 26 23 A LDA
01144A E521 27 0E E534 A BEQ
01145A E522 26 08 A BNE
01146A E523 26 05 E52F A BNE
01147A E524 CC 0004 A LDD

```

```

01148A E52D 20 0D E53C BPA
01149A E52F CC 0400 A CDSPFV
01150A E530 20 08 E53C BPA
01151A E531 9E 2F A CDSPN LDA
01152A E532 08 08 A BITB
01153A E533 27 02 E53C BEQ
01154A E534 1E 89 A EXG
01155A E535 34 50 A CDSPD PSHS
01156
01157A E536 84 07 A ANDA
01158A E537 C4 07 A ANDA
01159A E538 10 95 A CMPO
01160A E539 27 02 E543 BEQ
01161A E540 30 30 E4E6 BSR
01162A E541 30 30 E4E6 BSR
01163A E542 30 30 E4E6 BSR
01164A E543 30 30 E4E6 BSR
01165A E544 27 13 E565 BEQ
01166A E545 CE D0B3 A LDU
01167A E546 EF E4 A STU
01168A E547 C6 08 A LDB
01169A E548 34 04 A PSMS
01170A E549 A6 30 A CDSP3 LDA
01171A E54A 30 B1 E50E BSR
01172A E54B E7 48 A STB
01173A E54C 8D 40 E50E BSR
01174A E54D E7 C0 A STB
01175A E54E 6A E4 A DEC
01176A E54F 26 F2 E556 BNE
01177A E550 32 61 A LEAS
01178A E551 26 23 A CDSP50 LDA
01179A E552 26 16 E580 BNE
01180
01181A E553 EE E4 A LDU
01182A E554 AE 62 A LDX
01183A E555 CC 0008 A LDD
01184A E556 9D 30 A JSR
01185A E557 AE E4 A LDU
01186A E558 AE E4 A LDU
01187A E559 30 89 4000 A LEAX

```

```

01188A E57E CC 0008 A LDD
01189A E57F 30 9A A CDSPH JSP
01190A E580 EE E4 A CDSPH LDX
01191A E581 AE 62 A CDSPH LDX
01192A E582 AE 62 A CDSPH LDX
01193A E583 30 89 8000 A LEAX
01194A E584 CC 0008 A LDD
01195A E585 30 30 A JSP
01196A E586 35 D0 A RTSPUL
01197A E587 43 00 A PMSH
01198A E588 34 14 A CDSPH PMSH
01199A E589 35 06 E53A BSR
01200A E58A 30 01 A PULS
01201A E58B 30 01 A LEAX
01202A E58C 21 A STI2P
01203A E58D 30 A FCB
01204A E58E 30 A CDSPH COMA
01205A E58F 30 A CDSPH COMA
01206A E590 30 A CDSPH COMA
01207A E591 30 A CDSPH COMA
01208A E592 30 A CDSPH COMA
01209A E593 30 A CDSPH COMA
01210A E594 30 A CDSPH COMA
01211A E595 30 A CDSPH COMA
01212A E596 30 A CDSPH COMA
01213A E597 30 A CDSPH COMA
01214A E598 30 A CDSPH COMA
01215A E599 30 A CDSPH COMA
01216A E59A 30 A CDSPH COMA
01217A E59B 30 A CDSPH COMA
01218A E59C 30 A CDSPH COMA
01219A E59D 30 A CDSPH COMA
01220A E59E 30 A CDSPH COMA
01221A E59F 30 A CDSPH COMA
01222A E5A0 30 A CDSPH COMA
01223A E5A1 30 A CDSPH COMA
01224A E5A2 30 A CDSPH COMA
01225A E5A3 30 A CDSPH COMA
01226A E5A4 30 A CDSPH COMA
01227A E5A5 30 A CDSPH COMA
01228A E5A6 30 A CDSPH COMA
01229A E5A7 30 A CDSPH COMA
01230A E5A8 30 A CDSPH COMA

```

EXTEND A'S LOWER 4  
BITS TO 8BITS ON B.  
RETURNS A IS LOGICAL  
SHIFTED RIGHT 4.

CMPL #

X=FONT ADDR.

EACH COLOR LOADED.  
IF REVERSE IS DIRECTED  
EACH COLOR AND CHAR  
COLOR ARE EXCHANGED.  
SAVE COLORS, FONT ADDR  
AND DISPLAY ADDR.

IS REQUESTED COLOR SAME AS CURRENT COLOR?  
YES.  
H: COLOR

IS THE CHARACTER WIDTH  
SMALL?  
YES.

NO, MAKE LARGE CHAR.  
FONT ADDR RESTORE.

ORIGINAL FONT LOADED.  
EXTEND LOWER 4.

EXTEND HIGHER 4BITS.

END?  
NO, GO DO MORE.  
YES, RESTORE SP..  
MONO?  
YES.

NOT REQUEST

MSB PUT

LSB PUT

NOT REQUEST  
REQUEST BIT STRING STORE

RESTORE A.

ADD MARGIN  
IF MARGIN IS ZERO GO RETURN.

01221A E5B8 26	F8	E5B8	BNE	CDCLR	
01222A E5C1			DSRTS	VACOFF	
A E5C1 E7	D409	A	RTS	VACC	
01223A E5C4 39			RTS		
01224A E5C5 43			CDSONL	CDMA	
01225A E5C6 34	14	A	CDSOFL	PSHS	X, B
01226A E5C8 8D	E8	E5B2	BSF	CDSOFS	X, B
01227A E5CA 35	14	A	PULS	1, X	
01228A E5CC 80	01	A	LEAX	CDSOFS	
01229A E5CE 20	E2	E5B2	BRA		
01230					
01231A E5D0	E5B2	A	CDJMP	CDSOFS	OFF
01232A E5D2	E59A	A	FDB	CDSONMS	NORMAL
01233A E5D4	E593	A	FDB	CDSPVS	REVERSE
01234A E5D6	E581	A	FDB	CDSONS	ON
01235A E5D8	E5C6	A	FDB	CDSOFL	OFF
01236A E5DA	E530	A	FDB	CDSONML	NORMAL
01237A E5DC	E58F	A	FDB	CDSONML	REVERSE
01238A E5DE	E5C5	A	FDB	CDSONL	ON
01239			TTL	CONSL	PUT CODE
01241	D040	A	PCRORD	EQU	WDRI CR ORDER FLAG
01242	D041	A	PUTGET	EQU	PCRORD+1 PUT/GET
01243	D042	A	PUTIX	EQU	PUTGET+1
01244	D044	A	PUTFLD	EQU	PUTIX+2 EFASE FIELD FLAG
01245	D045	A	PUTEND	EQU	PUTFLD+1
01246					
01247					
01248					
01249					
01250					
01251A E5E0			FLGENT	PUT, GET, PUTGET	
A E5E0	86	A	GET	FCB	LDA #
A E5E1 4F		A	PUT	CLRA	
A E5E2 57	A1	A		STA	PUTGET
01253					
01254A E5E4 3E	D384	A	PUT000	EQU	DISPLAY STRING BEGIN
01255A E5E7 9F	A2	A	LDX		#.COMAR+4
01256A E5E9 0F	A0	A	STX		STRING PTR INIT.
		A	CLP	PCRORD	CLEAR CR ORDER FLAG FOR SUSPEND TO PUT.
01257		E5EB	PUT010	EQU	

01258A E5EE 6D	E745	A	JSP	PUTDA1	TAKE PUT CHAP
01259A E5EE 24	3D	E62D	BCC	PUT100	EXIST
01260A E5F0 0D	A1	A	TST	PUTGET	DATA END
01261A E5F2 1026	033A	E330	LENE	GET000	GET OPERATEFS INPUT
01262A E5F6 36	21	A	LDA	.CSCLT	SUSPEND TO PUT?
01263A E5F8 85	10	A	BITA	#CSLSUS	
01264A E5FA 27	2E	E62A	BEQ	PUT099	NO.
01265A E5FC 36	A0	A	LDA	PCRORD	DID WE DO CR ORDER?
01266A E5FE 27	2A	E62A	BEQ	PUT099	NO.
01267A E600 36	04	A	LDA	ICOUNT	
01268A E602 27	26	E62A	BEQ	PUT099	
01269A E604 C6	18	A	LDB	#ESC	LOAD ESC CODE
01270A E606 3E	05	A	LDX	IHEAD	GET ADDRESS OF KEYBUFFER
01271A E608 E1	80	A	PUT070	CMPEB	X+
01272A E60A 27	08	E614	BEQ	PUT080	YES, SUSPEND PUT.
01273A E60C 4A			DECA		MORE DATA IN BUFFER ?
01274A E60D 27	1B	E62A	BEQ	PUT099	NO, END PUT COMMAND.
01275A E60F 8D	F095	A	JSR	INCLBX	CORRECT PTR FOR BUFFER
01276A E612 20	F4	E608	BRA	PUT070	SEARCH MORE DATA.
01277A E614 8D	F03E	A	PUT080	JSP	IFBERRA
01278A E617 96	00	A	PUT030	LDA	..ABRT
01279A E619 26	0F	E62A	SNE	PUT099	BREAK WAS TYPED?
01280A E61B 36	04	A	LDA	ICOUNT	YES.
01281A E61D 27	F8	E617	BEQ	PUT090	
01282A E61F E6	9F	D005	A	LDB	[KHEAD]
01283A E623 C1	1B	A	CMPEB	#ESC	
01284A E625 27	03	E62A	BEQ	PUT099	
01285A E627 6D	F076	A	JSR	KEYINO	
01286		E62A	A	PUT099	EQU
01287A E62A 7E	EDE8	A	JMP	CURSET	
01288					
01289		E62D	A	PUT100	EQU
01290A E62D 05	21	A	LDB	.CSCLT	
01291A E62F C5	02	A	BITB	#CSLORD	
01292A E631 27	21	E654	BEQ	PUT105	ORDER DISABL
01293A E633 81	20	A	CMPEA	#1	
01294A E635 24	19	E650	BHS	PUT105	
01295A E637 34	02	A	PSHS	A	
01296A E639 8D	EDEF	A	JSR	CURCLR	
01297A E63C 35	02	A	PULS	A	

01298A E63E 81	16	A	CMPEA	#ESC	
01299A E640 1027	01B7	E7FB	LBEQ	ESCSEQ	
01300A E644 81	14	A	CMPEA	#S14	
01301A E646 24	12	E65A	BHS	PUT108	CURSOR ORDER
01302A E648 8E	E908	A	LDB	PUT090	
01303A E64B 48			LSLA		
01304A E64D AD	96	A	JSR	[A, X]	
01305A E64E 20	96	E5EE	BRA	PUT010	
01306					
01307A E650 81	7F	A	PUT105	CMPEA	#S7F
01308A E652 27	97	E5EB	BEQ	PUT010	
01309A E654 8D	22	E675	PUT105	BSP	PUTCHK
01310A E656 0F	1E	A	CLR	.CURST	
01311A E658 20	91	E5EE	BRA	PUT010	
01312					
01313A E65A 80	1C	A	PUT108	SUBA	#S1C
01314A E65C 2E	8D	E5EE	BMI	PUT010	CURSOR CODE?
01315A E65E 34	02	A	PSHS	A	NO, NO OPERATION.
01316A E660 8D	E77E	A	JSR	ETOC	SAVE
01317A E663 35	02	A	PULS	A	
01318A E665 48			LSLA		
01319A E668 8E	ECE4	A	LDX	#GETJCS	CURSOR ENTRY
01320A E669 AD	96	A	JSR	[A, X]	CURSOR MOVE
01321A E66B 8D	E787	A	JSR	CTOE	BUFFER PTR. SET
01322A E66E 6D	E1FC	A	JSR	EUPFTR	
01323A E671 0F	22	A	CLP	PUTCFG	
01324A E673 7E	E5EB	A	JMP	PUT010	
01325					
01326A E676 86	20	A	PUTSPC	LDA	#1
01327		E678	A	PUTCHR	EQU
01328A E678 34	16	A	PSHS		A, B, X
01329A E67A 47	9F	D047	A	STA	[CPAD] CHAR STORE
01330A E67E 9E	49	A	LDX	..BAAD	
01331A E680 D6	2E	A	LDB	..ACL	LOAD CURRENT ATTR
01332A E682 6D	84	A	TST	X	FILED?
01333A E684 2A	0C	E692	BPL	PUT110	NO. (THE FIELD TAG IS THE MSB.)
01334A E686 CA	80	A	ORL	#S0	
01335A E688 0D	22	A	TST	PUTCFG	
01336A E68A 27	06	E692	BEQ	PUT110	NO.
01337A E68C 8D	E6E4	A	JSP	PUTATR	ATTR SET ONE FIELD

## FM-7 サブシステム・モニタ ソース・リスト

01336A	E68F	D6	2E	A	LDB	..ACL	
01339A	E691			A	SKIP2		
	A E691	8C		A	FCB	#3C	CMPL #
01340A	E692	E7	84	A	PUT110	STB	SET
01341A	E694	BD	E51B	A	JSR	COSPE	ATTRIBUTE
01342A	E697	BD	E7BA	A	JSR	PUTINC	
01343A	E69A	86	FF	A	LDA	#9FF	
01344A	E69C	97	22	A	STA	PUTCFG	
01345A	E69E			A	RTSPUL	(A,B,X)	
	A E69E	35	96	A	PULS	A,B,X,PC	
01346					↑		
01347					↑↑↑↑↑↑↑↑↑↑↑↑↑↑		
01348					↑		
01349					↑ ERASE	FIELD	
01350					↑		
01351					↑↑↑↑↑↑↑↑↑↑↑↑↑↑		
01352					↑		
01353					↑		
01354	E6A0	0F	22	A	PUTOEL	EQU	↑
01355A	E6A2	ED	E77E	A	JSP	GLP	PUTCFG
01356A	E6A5	7E	E4EC	A	JSP	ETOC	
					JMP	ELKEY	
01357					↑		
01358					↑↑↑↑↑↑↑↑↑↑↑↑↑↑		
01359					↑		
01360					↑ BELL		
01361					↑		
01362					↑		
01363					↑		
01364	E6A8			A	GETBEL	EQU	↑
01365	E6A8			A	PUTOEL	EQU	↑
01366A	E6A8					BUZZON	
	A E6A8	B5	D403	A	RTA	BUZZER	
01367A	E6AB	39			PUTN0P	RTS	
01368					↑		
01369					↑↑↑↑↑↑↑↑↑↑↑↑↑↑		
01370					↑		
01371					↑ ERASE	ALL	
01372					↑		
01373					↑↑↑↑↑↑↑↑↑↑↑↑↑↑		
01374					↑		

014544	E715	97	46	A	STA
01455A	E715	07	45	A	STB
01456A	E719	3E	D03D	A	LDX
01457A	E71C	4F			CLRA
01458A	E71D	4C			INCA
01459A	E71E	E1	81	A	CMPE
01460A	E720	25	FB	E71D	BLO
01461A	E72C	E1	1F	A	CMPE
01462A	E724	24			BHS
01463A	E726	37	F	E71D	STA
01464A	E728	EC	1E	A	LDD
01465A	E72A	DD	4D	A	STD
01466A	E73C	ED	E375	A	JSR
01467A	E72F	20	90	E6C1	BRA
01468					*
01469					*****
01471					* REPEAT
01472					*
01473					*****
01474					*
01475			E731	A	PUT0FC EQU
01476A	E731	8D	12	E745	BSF
01477A	E733	25	D3	E738	BCE
01478A	E736	1F	89	A	TFR
01479A	E737	8D	0C	E745	BSR
01480A	E739	25	C	E708	BCE
01481A	E73E	5D			TSFB
01482A	E73C	27	06	E744	BEQ
01483A	E73E	8D	E678	A	PUT305 JSR
01484A	E741	5A			DECB
01485A	E742	26	FA	E73E	BNE
01486A	E744	39			PUT308 RTS
01487					*
01488			E745	A	PUT0AT EQU
01489A	E745	34	7	A	PSHS
01491A	E747	86	E683	A	PUT0AI EQU
01492A	E74A	26	06	E75A	LDA
01493A	E74C	36	01	A	LDA

```

01375      E2C  A  PUTACL  EQU  +
01376A E6AC 96 4F A  LDA  +,MDRG+2  KIND OF PLACE
01377A E8AE ED E2A0 A  JSR  ERASE1
01378A E6B1 0F 32 A  CLR  PUTCFG
01379A E6B3 33      RTS
01380      +
01381      ++++++
01382      +
01383      + NEW LINE  +
01384      +
01385      ++++++
01386      +
01387A E6B4 97 A0 A  PUTOCR STA  PCFORD  SET THE CR ORDER FLAG.
01388A E6B6 0F 46 A  CLR  .BPX  SET THE MOST LEFT POSITION.
01389A E6B8 96 21 A  LDA  .CSCSL
01390A E6BA 85 20 A  BITA  #CSSLF  IS THE AUTO LF ENABLE?
01391A E6BC 27 03 E6C1 A  BEQ  PUT550  NO.
01392      E6CE A  EQU
01393A E6BE 8D E7DE A  JSR  PUTDW
01394      E6C1 A  PUT550 EQU  +
01395A E6C1 0F 22 A  CLR  PUTCFG
01396A E6C3 7E 1FCF A  PUT560 JMP  BUFPTR
01397A E6C6 96 4D A  PUTOHM LDA  .MDRG  TOP Y COORDINATES
01398A E6C8 97 45 A  STA  .BPY
01399A E6CA 0F 46 A  CLR  .BPX
01400A E6CC 20 F3 E6C1 A  BRA  PUT550
01401      +
01402      +
01403      ++++++
01404      + +
01405      + START FIELD +
01406      + +
01407      ++++++
01408      +
01409      +
01410A E6CE 8D E6CE A  PUTOSF EQU  +
01411A E6D1 25 35 E745 A  JSR  PUTDAY  ATTR GET
01412A E6D3 84 3F A  BEB  PUT720
01413A E6D5 97 2E A  BSA  #-(AMFL+AMMD)-$FF01 NOT AMFL
01414A E6D7 8D 08 E6E4 A  STR  .ACL
01415A E6D9 8D 08 E6E4 A  PUTATR

```

```

01494A E74E 2B 11 E761 EMI
01495A E750 1A 01 SEC
01496A E752 RTSPUL
      A E752 35 30 A PULS
01497A E754 9E AC A PUTD8A LOX
01498A E756 AB 80 A LDA
01499A E758 3F AC A STX
01500A E75A 7A D383 DEC
01501A E75D 1C FE CLC
01502A E75F 90 A RTSPUL
      A E75F 35 30 A PULS
01503A E761 ED E121 A PUTD8A JSR
01504A E764 8E D384 A LOX
01505A E767 9F AC A STX
01506A E769 20 DC E747 ERA
01507 *
01508 *
01509 * *****
01510 * # BACKSPACE #
01511 * #
01512 * *****
01513 * #
01514 E76B A PUTDBS EQU
01515A E76B 9E 49 A LOX
01516A E76D AB 84 A LDA
01517A E76F 2B 0A E77B BHI
01518A E771 BD 06 E77E OBI
01519A E773 6D EA60 A JSR
01520A E776 80 0F E787 A BSR
01521A E778 BD E1FC A JSR
01522A E77B 0F 22 A PUTDBS CLR
01523A E77D 39 RTS
01524 *
01525A E77E 8E D045 A BTCC LOX
01526A E781 108E D058 A LDY
01527A E785 20 07 E78E LDX
01528A E787 BE D058 A CTDB OFA
01529A E78A 108E D058 A LDY
01530A E78E 05 08 A CTOEM LDB
01531A E790 7E 544A A JMP

```

01415A	E6D3	3E	43	A	LDX	.BAAD	
01416A	E6D8	A6	84	A	LDA	.X	
01417A	E6DD	8A	80	A	DRA	#AMFL	
01418A	E6DF	A7	84	A	STA	.X	
01419A	E6E1	0F	22	A	CLR	PUTCFG	
01420A	E6E3	39			RTS		
01421							
01422							
01423A	E6E4	34	06	A	PUTATR	EQU	
01424A	E6E6	3E	43	A	PSHS	A,B	
01425A	E6E8	A6	84	A	LDX	.E6AD	
01426A	E6EA	84	3F	A	ANDA	#3F	LOAD THE ATTR IN CURR. P
01427A	E6EC	A7	84	A	STA	.X	PICKUP USER'S ATTR FIELD
01428A	E6EE	1E	2E	A	CMPA	.ACL	SAME AS CURR ATTR.
01429A	E6F0	27	0C	E6FE	BEQ	PUTAT9	YES, DONT RESTORE.
01430A	E6F2	36	23	A	LDA	CSLCDS	MONO?
01431A	E6F4	26	08	E6FE	BNE	PUTAT9	YES, GO RETURN.
01432A	E6F6	36	2E	A	LDA	.ACL	
01433							
01434A	E6F8	A7	86	A	PUTAT5	EQU	
01435A	E6FA	E6	84	A	LDB	.X	
01436A	E6FC	2A	FA	E6F8	BPL	PUTAT5	
01437A	E6FE				PUTAT9	RTSPUL	(A,B)
	A E6FE	35	86	A	PULS	A,B,PC	
01438							
01439							
01440							
01441							
01442							
01443							
01444							
01445							
01446A	E700	8D	43	E745	BSR	PUTDAT	X CO-ORDINATE
01447A	E702	25	04	E708	BCR	PUT720	
01448A	E704	1F	89	A	TBR	A,B	
01449A	E706	8D	3D	E745	BSR	PUTDAT	Y CO-ORDINATE
01450A	E708	1025	F9C9	E0D5	PUT720	LBCS	
01451A	E70C	1E	89	A	EXG	A,B	
01452A	E70E	ED	E3D5	A	JSR	CACXCR	
01453A	E711	1024	F9C0	E0D5	LBCR	ERR62	ORDER ERROR

```

01532      +
01533      ++++++++
01534      + TAB +
01535      + +
01536      + +
01537      ++++++++
01538
01539      E793 A PUTOTB EQU
01540      E676 A JSR
01541      E796 36 46 A LDA
01542      E798 8D 03 E79D BSR
01543      E79A 27 F7 F7 BEQ
01544      E79C 39 RTS
01545
01546      E79D A TABCHK EQU
01547      E79D 06 21 A LDB
01548      E79F C5 04 A BITB
01549      E7A1 27 15 E788 BEQ
01550      E7A3 4D 12 TSTA
01551      E7A4 27 12 E78E BSR
01552      E7A6 1F 89 A TFF
01553      E7A8 54 EQ
01554      E7A9 54 LSRB
01555      E7AA 54 LSRB
01556      E7AB 84 A ANDA
01557      E7AD 8E E392 A LDX
01558      E7B0 86 86 A LDA
01559      E7B2 8E D0C4 A LDX
01560      E7B5 85 85 A LDA
01561      E7B7 39 RTS
01562      E7B8 4C TABCK9 INCA
01563      E7B9 39 RTS
01564      +
01565      ++++++++
01566      + INCREMENT +
01567      + +
01568      + +
01569      ++++++++
01570
01571      E7BA A PUTINC EQU

```

```

.BPX
.BPY
#.DMAP

.Y++
PUT750
-1,X
PUT750
.MDRG+2 I IND
-2,X
.MDRG FANGE
GETPRG
PUT590

```

```

*****
CHARACTER
*****

PUTDAT NO. OF REF
PUT720
A,B
PUTDAT CHAR
PUT720

PUT808
PUTCHF

PUT805

X
X
X
.COMAR+3
PUTDA2
..CNT

```

```

PUTDA3

X
X,PC
PUTIX
.X+
PUTIX
.COMAR+3

X
X,PC
CONT1
#.COMAR+4
PUTIX
PUTDA1

I
.EAAD
X
PUTOB8 FIELD TOP
BTDC
CSL
CTOB
BUFPTR
PUTCFG

#.BPY
#.CY
CTOBM
#.CY
#.BPY
#S
MOVE

```

```

I
PUTSPC
.BPX
TABCHI
PUTOTB

I
.CSCTL
#CSLTAB
TABCK3

TABCK3
A,B

#S07
#BAS
A,X
#.TABST
B,X

```

```

01572A E7BA 0C 46 A INC .BPX
01573A E7BC 96 46 A LDA .BPX
01574A E7BE 91 36 A CMPA .LNC
01575A E7C0 24 16 E7D3 BHS PUTIN5 FIGHT OVER
01576 E7C2 A PUTIN2 EQU I
01577A E7C2 9E 47 A LDX I
01578A E7C4 30 01 A LEAX 1,X INCREMENT CHAR PTR
01579A E7C6 9F 47 A STX I
01580A E7C8 3E 49 A LDX I
01581A E7CA 30 01 A LEAX 1,X INCREMENT ATTR PTR
01582A E7CC 9F 49 A STX I
01583A E7CE DC 39 A LOD I
01584A E7D0 D3 48 A ADDD I
01585A E7D2 D0 48 A STD I
01586A E7D4 39 A RTS INCREMENT VRAM PTR
01587 E7D5 A PUTIN3 EQU I
01588A E7D5 7E E1FC JMP I BUFPTR
01589 A I
01590A E7D8 0F 46 A PUTIN5 CLR .BPX DOWN
01591A E7DA 96 21 A LDA .CSCTL
01592A E7DC 28 F7 E7D5 BHI PUTIN3
01593 A I
01594 A I
01595 A I
01596 A I DOWN I
01597 A I
01598 A I
01599 A I
01600 A I
01601A E7DE 0C 45 A PUTDW EQU I
01602A E7E0 96 45 A INC .BPY
01603A E7E2 91 4E A LDA .BPY
01604A E7E4 25 EF E7D5 CMPA .MDRG+1
01605 A BLO PUTIN3 INNER
01606 A I
01607A E7E6 9F 4F A LDA .MDRG+2 KIND OF WINDOW
01608A E7E8 31 01 A CMPA #1
01609A E7EA 27 42 E82E BEQ SCROLL
01609 A I
01610A E7EC 96 21 A LDA .CSCTL
01611A E7EE 85 08 A BITA #CSLPAG PAGE LIMIT WAIT ?

```

```

01612A E7F0 27 06 E7F8 BEQ PUTD70 NO.
01613A E7F2 A BITA BUZZON
01614A E7F2 85 D403 A BITA BUZZER
01615A E7F5 ED F06E A JSR KEYGET
01616A E7F8 7E ED E8AC A PUTD70 JMP PUTOTCL
01617 0039 A ESCRER EQU I ERASE KEY BUFFER
01618 0028 A ESCRLC EQU I# KEY LOCK
01619 0022 A ESCRUL EQU I# KEY UNLOCK
01620 0067 A ESCBUF EQU I#G+$20 BUFFER MODE
01621 0068 A ESCBNF EQU I#H+$20 NONBUFFER MODE
01622A E7F8 8D E745 A ESCSEQ JSR PUTDAT GET SECONDARY CHAR.
01623A E7FE 25 35 E82E BNE ESCSE9
01624A E800 81 39 A CMPA #ESCIEF ERASE?
01625A E802 26 05 E809 BNE ESCSE2 NO.
01626A E804 8D F03E A JSR I#SFERA DO BUFFER ERASE.
01627A E807 20 22 E82E BRA ESCSE9
01628A E809 81 23 A ESCSE2 CMPA #ESCRLC LOCK?
01629A E80B 26 05 E813 BNE ESCSE3 NO.
01630A E80D 96 FF A LDA #FFF SET LOCK.
01631A E80F 97 02 A STA I#LOCK
01632A E811 20 18 E82E BRA ESCSE9
01633A E813 81 22 A ESCSE3 CMPA #ESCRLC UNLOCK?
01634A E815 26 04 E81E BNE ESCSE4 NO.
01635A E817 0F 02 A CLR I#LOCK SET UNLOCK
01636A E819 20 10 E82E BRA ESCSE3
01637A E81B A ESCSE4 EQU I
01638A E81B 81 67 A CMPA #ESCBUF
01639A E81D 26 06 E825 BNE ESCSE5
01640A E81F 96 FF A LDA #FFF SET BUFFER MODE
01641A E821 27 03 A STA I#BFLL6
01642A E823 20 06 E82E BRA ESCSE9
01643A E825 81 68 A ESCSE5 CMPA #ESCBNF NON BUFFER?
01644A E827 26 02 E82E BNE ESCSE6 NO, CHECK OTHER.
01645A E829 0F 03 A CLR I#BFLL6 SET NON BUFFER MODE
01646A E82B A ESCSE6 EQU I
01647A E82B A ESCSE9 EQU I
01648A E82B A ESCSE9 EQU I
01649A E82B 7E E9EB A JMP PUTD10

```

```

01650 *****
01651 I
01652 I SCROLL I
01653 I
01654 *****
01655 I
01656 I
01657 I
01658 A I
01659A E82E 96 4E A SCROLL EQU I
01660A E830 D6 4E A LDA .MDRG+1
01661A E832 5C 7D A LOD .MDRG+1
01662A E834 DD 70 A INC8
01663A E836 31 44 A STD .CVTOP
01664A E838 26 32 E86E A CMPA .DMAP+7 IS THE DISPLAY FULL SCROLL?
01665A E83A D8 4D A BNE SCROLL NO.
01666A E83C 0F 40 A LOD .MDRG IS THE TOP OF SCROLL WINDOW SAME AS THE TOP OF DISPLAY?
01667 A I
01668A E83E 34 01 A PSHS CC
01669A E840 26 2E E86D BNE SCROLL NO.
01670A E842 96 38 A LDA .FS DISPLAY PF?
01671A E844 27 11 E854 BEQ SCROLL NO.
01672A E846 DC 7D A LOD .CVTOP YES, ERASE THE LAST LINE OF DISPLAY.
01673A E848 C3 0202 A ADDD #S0202
01674A E84A DD 7D A STD .CVTOP
01675A E84C 8D E203 A JSR ERSVRM
01676A E84E DC 7D A LOD .CVTOP
01677A E850 33 0202 A SUBD #S0202
01678A E852 DD 7D A STD .CVTOP
01679 A I
01680A E854 8D E203 A JSR ERSVRM ERASE LAST LINE BEFORE SCROLLING.
01681A E856 DC 1F A LOD .VOFST
01682A E858 D3 38 A ADDD .VLU
01683A E85E A I
01684A E860 85 D403 A VACON
01685A E862 DD 1F A BITA
01686A E864 DD 1F A STD VACC
01687A E866 FD D40E A STD VOFFST
01688A E868 A I
01689A E86A 87 D403 A STA VACC
01690A E86C DD E00E A JSR INDICT
01691A E86E 20 51 E86C BRA SCROLL15

```

# FM-7 サブシステム・モニタ ソース リスト

```

01689          EQU 01
01690          A SCF004 EQU CC
01691A E86B 34 01 A PSMS CC
01692A E86D 40 A SCF005 DECA
01693A E86E 91 4D A CMPA .MORG
01694A E870 77 5A E80C BEQ SCF090
01695A E872 96 4E A LDA .MORG+1
01696A E874 8D E221 A JSR BVADL
01697A E877 1F 01 A TFR D,X SOURCE END ADDRESS
01698A E879 96 4D A LDA .MORG
01699A E87B 1F E221 A JSR BVADL
01700A E87E 1F 02 A TFR D,Y DESTINATION TOP
01701A E880 D3 3E A ADDO .VLU
01702A E882 1F 93 A TFC D,U SOURCE TOP
01703A E884 A VACON
01704A E887 96 23 A EITA VACC
01705A E889 26 20 E84B LDA CSLCOS MONO?
01706A E88E 34 10 A BNE MNSCRL YES.
01707A E89D 1F 31 A PSMS X SOURCE END
01708A E89F EC 89 8000 A MNSCRL LDD U,X LOAD TOP OF SOURCE TO X.
01709A E8A3 ED A9 8000 A STD .,VG,X
01710A E8A7 EC 89 4000 A LDD .,VR,X
01711A E8AB ED A9 4000 A STD .,VR,X
01712A E8AF EC 81 A LDD .,Y++
01713A E8B1 ED A1 A STD .,Y++
01714A E8B3 AC E4 A CMPX +S
01715A E8B5 25 0E E88F E3 BLO MNSCRL
01716A E8B7 32 52 A LEAS +S
01717A E8B9 20 0E E88B BRA SCF013
01718A E8BB 30 89 8000 A MNSCRL LEAX .,VG,X
01719A E8BD 31 A9 8000 A LEAY .,VG,Y
01720A E8BF 33 C9 8000 A LEAU .,VG,U
01721A E8C1 8D 38 E8F1 BSR COPLDP
01722A E8C3 A E8E9 87 D403 A SCF013 STA VACC
01723          A SCF015 LDA .MORG DESTINATION TOP ADDRESS
01724A E8C6 96 4D A JSP BADRL
01725A E8C8 6D E208 A JSP X,Y SAVE
01726A E8C1 34 30 A PSMS

```

```

01727A E8C3 96 4E A LDA .MORG+1 SOURCE END ADDRESS
01728A E8C5 6D E208 A JSR BADRL
01729A E8C8 35 40 A PULS U LOAD DESTINATION TOP ADDRESS.
01730A E8CA 34 20 A PSMS Y SAVE ATTP'S SOURCE END ADDRESS.
01731A E8CC 31 C4 A LEAY U SET Y DESTINATION TOP ADDRESS.
01732A E8CE D6 36 A LDB .,LNC
01733A E8D0 33 C5 A LEAU S,U SET U SOURCE TOP ADDRESS
01734A E8D2 8D 10 E8F1 BSR COPLDP COPY THE CHAR.
01735A E8D4 35 30 A PULS X,Y LOAD SOURCE ADDRESS TO X.
01736          A LDB .,LNC LOAD DESTINATION TOP ADDRESS TO Y.
01737A E8D6 D6 36 A LDB .,LNC
01738A E8D8 33 A5 A LEAU S,Y MAKE SOURCE TOP ADDRESS.
01739A E8DA 8D 15 E8F1 BSR COPLDP COPY THE ATTP.
01740          A SCF090 EQU +
01741          A SCF090 EQU +
01742A E8DC DC 7D A LDD .,CVTDP
01743A E8DE 4A DECA DECA
01744A E8DF 5A DECA DECA
01745A E8E0 97 45 A STA .BPPY
01746A E8E2 DD 7D A STD .,CVTDP
01747A E8E4 35 01 A PULS CC
01748A E8E6 27 03 E8EB BEQ SCF095
01749A E8E8 6D E203 A JSP ERSVPM
01750A E8EA 6D E331 A SCF095 JSP ERSBUF
01751A E8EE 7E E205 A JNP ASNFLD ASSUME FIELD.
01752          A SCF090 EQU +
01753          A SCF090 EQU +
01754          A SCF090 EQU +
01755          A SCF090 EQU +
01756          A SCF090 EQU +
01757          A SCF090 EQU +
01758          A SCF090 EQU +
01759          A SCF090 EQU +
01760          A SCF090 EQU +
01761          A SCF090 EQU +
01762          A SCF090 EQU +
01763          A SCF090 EQU +
01764A E8F1 34 10 A PSMS X
01765A E8F3 37 16 A COPLDP PULS D,X
01766A E8F5 ED A4 A STD D,Y

```

```

01767A E8F7 AF 22 A STX 2,Y
01768A E8F9 37 16 A PULS D,X
01769A E8FB ED C4 A STD 4,Y
01770A E8FD AF 26 A STX 6,Y
01771A E8FF 31 28 A LEAY S,Y
01772A E901 11A3 E4 A CMPU +S
01773A E904 25 ED E8F3 BLO COPLDP
01774A E906 30 90 A RTSPUL X
01775          A E906 35 90 A PULS X,PC
01776          A E906 35 90 A PULS X,PC
01777          A E906 35 90 A PULS X,PC
01778          A E906 35 90 A PULS X,PC
01779A E908 A E908 A PUTORD EQU +
01780A E90A A E90A A PUTORD EQU +
01781A E90C A E90C A PUTORD EQU +
01782A E90E A E90E A PUTORD EQU +
01783A E910 A E910 A PUTORD EQU +
01784A E912 A E912 A PUTORD EQU +
01785A E914 A E914 A PUTORD EQU +
01786A E916 A E916 A PUTORD EQU +
01787A E918 A E918 A PUTORD EQU +
01788A E91A A E91A A PUTORD EQU +
01789A E91C A E91C A PUTORD EQU +
01790A E91E A E91E A PUTORD EQU +
01791A E920 A E920 A PUTORD EQU +
01792A E922 A E922 A PUTORD EQU +
01793A E924 A E924 A PUTORD EQU +
01794A E926 A E926 A PUTORD EQU +
01795A E928 A E928 A PUTORD EQU +
01796A E92A A E92A A PUTORD EQU +
01797A E92C A E92C A PUTORD EQU +
01798A E92E A E92E A PUTORD EQU +
01799          A E92E A E92E A PUTORD EQU +
01800          A E92E A E92E A PUTORD EQU +
01801          A E92E A E92E A PUTORD EQU +
01802          A E92E A E92E A PUTORD EQU +
01803          A E92E A E92E A PUTORD EQU +
01804          A E92E A E92E A PUTORD EQU +
01805          A E92E A E92E A PUTORD EQU +

```

```

01806          A E92E A E92E A PUTORD EQU +
01807          A E92E A E92E A PUTORD EQU +
01808          A E92E A E92E A PUTORD EQU +
01809          A E92E A E92E A PUTORD EQU +
01810          A E92E A E92E A PUTORD EQU +
01811          A E92E A E92E A PUTORD EQU +
01812A E930 0F 22 A GET000 EQU CLR
01813A E932 8D EDE8 A JSR
01814          A E932 8D EDE8 A JSR
01815          A E932 8D EDE8 A JSR
01816          A E932 8D EDE8 A JSR
01817A E935 9E 54 A LDX
01818A E937 9E 54 A GET050 LDX
01819A E939 84 EF A ANDA
01820A E93B A7 80 A STA
01821A E93D 9C 56 A CMPLX
01822A E93F 25 F6 E937 BLO
01823          A E937 25 F6 E937 BLO
01824          A E937 25 F6 E937 BLO
01825A E941 8D E941 A GET100 EQU JSP
01826A E943 86 01 A LDA
01827A E945 97 1D A STA
01828A E947 8D EE07 A JSP
01829A E949 8D FD6E A JSP
01830A E94B 34 02 A PSMS
01831A E94D 8D EDED A JSP
01832A E94F 35 02 A PULS
01833A E951 D8 00 A LDB
01834A E953 1026 F603 E934 LBNB
01835A E955 81 7F A CMPA
01836A E957 26 05 E964 BNE
01837A E959 8D EE03 A JSP
01838A E95B 20 0D E941 BRA
01839A E95D 81 20 A GET110 CMPA
01840A E95F 25 05 E96D JSP
01841A E961 8D E93A A JSP
01842A E963 8D E941 A BRA
01843A E965 48 04 E941 GET120 LSLA
01844A E967 8E EC4C A LOX
01845A E969 8E 96 A JSR

```

```

01846A E973 20 CC E941 BRA
01847          A E941 BRA
01848          A E941 BRA
01849          A E941 BRA
01850          A E941 BRA
01851          A E941 BRA
01852          A E941 BRA
01853          A E941 BRA
01854A E975 96 4E A GETFNG EQU
01855A E977 96 4E A JSP
01856A E979 96 4E A JSP
01857A E97B 96 4E A STY
01858A E97D 96 4E A LDA
01859A E97F 8D E208 A JSP
01860A E981 9F 50 A STX
01861A E983 109F 54 A STY
01862A E985 39 A GETNOP RTS
01863          A E985 39 A GETNOP RTS
01864          A E985 39 A GETNOP RTS
01865          A E985 39 A GETNOP RTS
01866          A E985 39 A GETNOP RTS
01867          A E985 39 A GETNOP RTS
01868          A E985 39 A GETNOP RTS
01869          A E985 39 A GETNOP RTS
01870          A E985 39 A GETNOP RTS
01871A E98A 9E 5A A GETD EQU
01872A E98C 109E 5C A LDX
01873A E98E 56 A4 A LDY
01874A E990 C5 10 A LDB
01875A E992 27 04 E999 BEQ
01876A E994 85 A SUZZON
01877A E996 39 D403 A RTS
01878          A E996 39 D403 A RTS
01879A E998 0D 33 A GETD0 TST
01880A E99A 26 35 E9D2 BNE
01881A E99C A7 84 A GETD1 STA
01882A E99E 7E 40 A ORS
01883A E9A0 E7 A4 A STB
01884A E9A2 8D E518 A JSR

```

```

01885A E9A6 8D EA44 A JSP
01886A E9A8 24 14 E9BF A ECC
01887A E9AA 96 4F A LDA
01888A E9AC 4A 0F A DECA
01889A E9AE 28 0F E9BF A BNE
01890A E9B0 6D E8CE A JSP
01891A E9B2 03 A6 A COM
01892A E9B4 0F 46 A CLR
01893A E9B6 DC 45 A LDD
01894A E9B8 DD 58 A STD
01895A E9BA 7E E1FC A JNP
01896A E9BC 39 A GETD4 RTS
01897A E9BE 8D E3C3 A GETD2 JSP
01898A E9C0 E6 A4 A LDB
01899A E9C2 2A 03 E9C9 BPL
01900A E9C4 30C5 03 E9C9 SUZZON
01901A E9C6 65 D403 A BITA
01902A E9C8 9C 45 A GETD3 STA
01903A E9CA 23 EF E9EE A CMPLX
01904A E9CC 7E E737 A BLS
01905          A E9CC 7E E737 A JMP
01906          A E9CC 7E E737 A JMP
01907A E9D0 0F A6 A GETINS EQU
01908A E9D2 31 21 A CLR
01909A E9D4 60 30 A LEAY
01910A E9D6 27 08 E9E2 A GET110 TST
01911A E9D8 60 A0 A BEQ
01912A E9DA 2A F8 E9D6 TST
01913A E9DC 8D A0 A STA
01914A E9DE 85 D403 A SUZZON
01915          A E9DE 85 D403 A BITA
01916A E9E0 9E 58 A RTS
01917A E9E2 9E 58 A GET120 EQU
01918A E9E4 34 10 A LDX
01919A E9E6 9E 5A A PSMS
01920A E9E8 109E 5C A LDB
01921A E9EA 1F 03 A GET130 EQU
01922A E9EC 84 84 A TFR
01923A E9EE 84 84 A LDA

```



\*\*\*\*

PUTEND  
GTFLD+1#  
PUTCFG  
CURSET

DATA TAG

GTTPA  
+X ATTR  
#-AMMD-\$FF01 REMOVE

+X+

GTBTMA

GET050

#

CEADR

#1

CURTM

CURGO

LEYGET

A

CURSCL

A

..ABRT ABDRT?

CNDCTL YES.

#37F DELETE

GET110

GETDEL

GET100

#1

GET120

GETDT DATA

GET100

#GETJMP CONTROL KEY

[A,X]

GET100

#

.MDRG+1

EADRL

GTBTMC CHAR BTM

GTBTMA ATTR BTM

.MDRG

EADRL

GTTPC

GTTPA

#

.CEAD

.CAAD

+Y

#AMPT

GETDT0

BUZZER

..INS INSERT ?

GETINS YES

+X

#AMMD

+Y

CDSPC

CSR

GETDT2

.MDRG+2

GETDT2

PAGE

SCROLL

GTSCL SET SCROLL FLAG FOR INSERT MODE.

.BPX

.BPY

.CY

EUPPTR

CEADR

+Y

GETDT3

BUZZER

.CY

.BPY

GETDT4

CTOE

#

GTSCL

1,Y PRE INC

+X+ NULL SEARCH

GET120 FOUND

+Y+ ANFL IS M56

GET110 NOT FIELD

BUZZER

#

.CY

X

.CEAD

.CAAD

#

D,U CHAR,ATTR SAVE

+X

```

01323A ESEF E6 44 A LDE +Y
01324A ESF1 C4 7F A ANDB #-ANFL-$FF01 REMOVE FIELD
01325A ESF2 34 06 A PSHS A,B STORE POS DATA SAVE
01326A ESF5 1F 30 A TFR U,D
01327A ESF7 80 44 E930 BSR GETDT1
01328A ESF9 8D ESC3 A JSR CEADP
01329A ESFC 35 06 A PULS A,B
01330A ESFE 4D TSTA
01331A ESFF 26 EA E9EB BNE GETI30
01332A EA01 35 06 A PULS D NOT NULL

01334A EA03 00 58 A STD .CY RESTORE CURSOR POS.
01335A EA05 00 A6 A TST GTSCRL EXECUTED SCROLL?
01336A EA07 27 38 EA44 BEQ CSF NO, CURSOR TO RIGHT.
01337A EA09 81 4D A CMPA .MDRG YES, IS THE CURSOR ON THE START LINE
01338A EA0B 27 04 EA11 BEQ GETI40 YES.
01339A EA0D 6A 56 A DEC .CY NO, CURSOR TO UP.
01340A EA0F 20 33 EA44 BSR CSF CURSOR TO RIGHT.
01341A EA11 0F 59 A GETI40 CLR .CX CURSOR TO TOP OF WINDOW.
01342A EA13 39 A RTS

01343
01344
01345
01346
01347
01348
01349
01350
01351A EA14 0C EA14 A CSRL EQU +
01352A EA16 36 59 A INC .CX
01353A EA18 31 36 A LDA .CX
01354A EA1A 2D 02 EA1E CMPA .LNC
01355A EA1C 0F 59 A BLT CSRL90
01356A EA1E 39 A CLR .CX OVER
01357
01358
01359
01360
01361
01362
01363
01364
01365A EA1F 0A 59 A CSLL EQU +
01366A EA21 2A 05 EA28 SPL DEC .CX
01367A EA23 96 36 A SPL CSLL90
01368A EA25 4A LDA .LNC
01369A EA27 37 A STA DECA
01370A EA28 39 A CSLL90 RTS .CX

01371
01372
01373
01374
01375
01376
01377
01378
01379A EA29 0A 58 A CSUL EQU +
01380A EA2B 96 58 A DEC .CY
01381A EA2D 91 4D A LDA .CY
01382A EA2F 2C 05 EA3E CMPA .MDRG
01383A EA31 96 4E A BSR CSUL90
01384A EA33 4A 58 A LDA .MDRG+1
01385A EA35 37 A STA DECA
01386A EA36 39 A CSUL90 RTS .CY

01387
01388
01389
01390
01391
01392
01393
01394
01395A EA37 0C 58 A CSDL EQU +
01396A EA39 36 58 A INC .CY
01397A EA3B 91 4E A LDA .CY
01398A EA3D 2C 04 EA43 CMPA .MDRG+1
01399A EA3F 96 4D A BLT CSDL90
02000A EA41 97 58 A LDA .MDRG
02001A EA43 39 A STA .CY
02002
02003
02004
02005
02006
02007
02008
02009
02010A EA44 0C 59 A CSF EQU +
02011A EA46 96 59 A INC .CX
02012A EA48 91 36 A LDA .CX
02013A EA4A 2D 11 EA5D CMPA .LNC
02014A EA4C 0F 59 A BLT CSRL90
02015A EA4E 0C 58 A CLR .CX
02016A EA50 36 58 A INC .CY
02017A EA52 91 4E A LDA .CY
02018A EA54 2D 07 EA50 CMPA .MDRG+1
02019A EA56 96 4D A BLT CSRL90
02020A EA58 97 58 A LDA .MDRG
02021A EA5A 1A 01 A STA .CY
02022A EA5C 39 A SEC
02023
02024
02025A EA5D 1C 59 A CSRL90 EQU +
02026A EA5F 39 FE A CSRL90 CLC
02027
02028
02029
02030
02031
02032
02033
02034
02035A EA60 0A 59 A CSL EQU +
02036A EA62 2A F9 EA5D DEC .CX
02037A EA64 96 36 A SPL CSRL90
02038A EA66 4A A LDA .LNC
02039A EA67 37 59 A STA .CX
02040A EA69 0A 58 A DEC .CY
02041A EA6B 96 58 A LDA .CY
02042A EA6D 91 4D A CMPA .MDRG

```

# FM-7 サブシステム・モニタ ソース・リスト

```

02043A EA6F 2C EC EASD BGE CSL90
02044A EA71 36 4E A LDA .MDRG+1
02045A EA73 4A DECA
02046A EA74 97 58 A STA .CY
02047A EA76 1A 01 SEC
02048A EA78 39 RTS
02049
02050
02051
02052 *****
02053 # ENCODE CO-ORDINATE #
02054 *****
02055 *****
02056 *****
02057 EA79 A GETENC EQU #
02058A EA79 1F 10 A TFR X,D CHAR
02059A EA7B 33 50 A SUBD GTTOPC OFFSET
02060A EA7D 20 04 EA83 BRA GETEN
02061
02062 EA7F A GETENA EQU #
02063A EA7F 1F 20 A TFR Y,D
02064A EA81 93 54 A SUBD GTTOPA OFFSET
02065
02066 EA83 A GETEN EQU #
02067A EA83 9E 40 A LDX .MDRG
02068A EA85 9F 58 A STX .CY TOP Y,DESTFOY .CX
02069
02070A EA87 93 35 A GETEN5 SUBD .LNCOD
02071A EA89 26 04 EA8F BML GETEN3
02072A EA8E 0C 58 A INC .CY
02073A EA8D 20 F8 EA87 BRC GETEN5
02074
02075A EA8F 03 36 A GETEN3 ADDD .LNCOD
02076A EA91 07 59 A STB .CX
02077A EA93 39 RTS
02078
02079 *****
02080 # HOME #
02081 *****
02082 *****
02083 *****

```

```

02084
02085 EA94 A CSHOME EQU #
02086A EA94 103E 54 LDY GTTOPA ATTR
02087A EA97 86 80 A CSHOM0 LDA #AMFL
02088A EA99 C6 10 A LDB #AMPT
02089A EA9E A5 A4 A CSHOM5 BITA .Y
02090A EA9D 27 04 EA93 BEQ CSHOM7
02091A EA9F E5 A4 A BITB
02092A EAA1 27 0A EAAD BEQ CSHOM9
02093A EAA3 31 21 A CSHOM7 LEAY 1,Y
02094A EAA5 103C 56 A CMPY GTBTMA
02095A EAA8 25 F1 EA9B BLO CSHOM5
02096A EAAA 103E 54 A CSHOM8 LDY GTTOPA
02097A EAAD 80 D0 EA7F CSHOM9 BSR GETENA
02098A EAAF 39 RTS
02099
02100 *****
02101 # FORWARD FIELD #
02102 *****
02103 *****
02104 *****
02105 *****
02106 EA80 A GETFFD EQU #
02107A EA80 8D EA44 A JSP CSR RIGHT
02108A EA83 8D E3C3 A JSP CBADR
02109A EA86 20 DF EA97 BRA CSHOM0
02110
02111 *****
02112 # BACKWARD FIELD #
02113 *****
02114 EA88 A GETBFD EQU #
02115A EA88 8D EA50 A JSP CSL LEFT
02116A EA8B 8D E3C3 A JSP CBADR
02117
02118 EA8E A GETBHM EQU #
02119 *****
02120 # SEARCH UNPROTECTED FIELD #
02121 # FOR BACK #
02122 *****
02123A EABE CC 8010 A LDD #AMFL1256+AMPT

```

```

02124A EAC1 A5 A4 A GETEH5 BITA .Y
02125A EAC3 27 04 EA03 BEQ GETEH7 NO FIELD
02126A EAC5 E5 A4 A BITB
02127A EAC7 27 64 EAAD BEQ CSHOM9 DEFINE
02128A EAC9 31 3F A GETEH7 LEAY -1,Y
02129A EACB 103C 54 A CMPY GTTOPA
02130A EACE 24 F1 EAC1 BHS GETEH5
02131A EAD0 20 D3 EAAA BRA PHISICAL HOME
02132
02133 *****
02134 *****
02135 # CLEAR #
02136 *****
02137 *****
02138 *****
02139 *****
02140A EAD2 96 4F A GETCLR EQU #
02141A EAD4 8D E2A0 A JSP EFASE1
02142A EAD7 0C 45 A LDD .EPY
02143A EAD9 0D 58 A STD .CY
02144A EADB 86 0C A LDA #2
02145A EADD 7E EC42 A JMP GETCF1
02146
02147 *****
02148 *****
02149 # INSERT #
02150 *****
02151 *****
02152 *****
02153 *****
02154A EAE0 03 33 A INSERT EQU #
02155A EAE2 26 04 EA03 COM .INS
02156A EAE4 A EAE8 B5 0400 A INSOFF INSEF5
02157A EAE7 39 0400 A STA STA INSLED
02158A EAE8 B5 0400 A INSEF5 INSON
02159A EAE9 39 0400 A INSEF5 INSON
02160 *****

```

```

02161 *****
02162 *****
02163 # EFASE FIELD #
02164 *****
02165 *****
02166 *****
02167 EAEC A ELK.EY EQU
02168A EAEC DC 58 A LDD
02169A EAEE 34 06 A PSHS
02170A EAF0 9E 5A A LDX
02171A EAF2 103E 5C A LDY
02172A EAF5 E8 A4 A LDB
02173A EAF7 CA 40 A ORL
02174A EAF9 E7 A4 A STB
02175 EAFB A ELK.EY2 EQU
02176A EAFB A6 84 A LDA
02177A EAFD 27 08 EB07 BEQ
02178A EAFD 4F 84 A CLRA
02179A EB00 A7 47 A STA
02180A EB02 E5 A4 A LDB
02181A EB04 E0 E518 A JSR
02182 EB07 A ELK.EY4 EQU
02183A EB07 E0 EA44 A JSR
02184A EB0A E0 E3C3 A JSR
02185A EB0D E5 A4 A LDB
02186A EB0F 2A EA A EAFB
02187A EB11 3C 47 A CMPL
02188A EB13 23 07 EB1C BLS
02189A EB15 EC E4 A LDD
02190A EB17 D0 45 A STD
02191A EB19 6D E1FC A JSR
02192 EB1C A ELK.EY6 EQU
02193A EB1C 25 06 A PULS
02194A EB1E D0 58 A STD
02195A EB20 39 RTS
02196
02197 *****
02198 *****
02199 *****
02200 *****
02201 *****

```

```

02202
02203A EB21 96 36 A GETOUT EQU
02204A EB23 D6 58 A LDA
02205A EB25 D1 40 A CMBP
02206A EB27 27 2C EB55 BEQ
02207A EB29 40 21 A NEGA
02208A EB2A 0A 21 A SLIP1
02209A EB2A 0A 21 A FCB
02210
02211 EB2B A TAB EQU
02212A EB2B 4F 33 A CLPA
02213A EB2C D5 33 A TABDUP LDB
02214A EB2E 34 06 A PSHS
02215A EB30 0F 33 A CLP
02216A EB32 46 20 A TAB10 LDA
02217A EB34 E6 E4 A LDB
02218A EB36 27 04 EB3C BEQ
02219A EB38 9E 5A A LDX
02220A EB3A A6 85 A LDA
02221A EB3C E0 E98A A TAB15 JSR
02222A EB3F 8D E3C3 A JSR
02223A EB42 36 59 A LDA
02224A EB44 6D E73D A JSR
02225A EB47 26 08 EB51 SNE
02226A EB49 9E 5C A LDX
02227A EB4B E6 84 A LDB
02228A EB4D C5 90 A BITB
02229A EB4F 27 E1 EB32 BEQ
02230A EB51 35 06 A TAB90 PULS
02231A EB53 D7 33 A STB
02232A EB55 39 GETOUT3 RTS
02233
02234 *****
02235 *****
02236 *****
02237 *****
02238 *****
02239 *****
02240 *****
02241A EB56 0F EB56 A CSFW EQU
02242A EB56 0F A5 A CLP

```

```

02242A EB58 3E 5A A LDX
02243A EB5A 103E 5C A LDY
02244A EB5D 8D 16 EB75 CSRW10 BSR
02245A EB5F 25 FC EB5D BCS
02246
02247A EB61 8D 12 EB75 CSRW20 BSR
02248A EB63 24 FC EB61 BCC
02249A EB65 30 1F A LEAX
02250
02251 *****
02252 *****
02253A EB67 9C 52 A CSRW30 CMPL
02254A EB69 25 07 EB72 BLO
02255A EB6B 96 40 A LDA
02256A EB6D 97 58 A STA
02257A EB6F 0F 59 A CLP
02258A EB71 39 06 A RTS
02259A EB72 7E EA79 A CSRW35 JMP
02260
02261 EB75 A CSRW EQU
02262A EB75 0D A5 A TST
02263A EB77 26 0F EB88 BNE
02264A EB79 3D 11 EB88 BSR
02265A EB7E 30 01 A LEAX
02266A EB7D 31 21 A LEAY
02267A EB7F C6 80 A LDB
02268A EB81 E5 A4 A BITB
02269A EB83 27 02 EB87 BEQ
02270A EB85 0C A5 A INC
02271A EB87 39 06 A CSRW55 RTS
02272A EB89 32 62 A CSRW59 LEAS
02273A EB8A 20 D8 EB87 BFA
02274
02275 EB8C A CSWC EQU
02276A EB8C A6 84 A LDA
02277A EB8E CE EBDC A LDB
02278A EB91 E6 C0 A LDB
02279A EB93 A1 C1 A CSWC1 CMPA
02280A EB95 25 07 EB9E BLO

```



# FM-7 サブシステム・モニタ ソース・リスト

```

02396 *****
02397 * CANCEL *
02398 *
02399 *****
02400 *
02401 *
02402 EC3E EC41 A GETCAN EQU #1
02403A EC3E 86 01 A LDA
02404A EC40 21 A SKIP1
A EC40 21 A FCB $21
02405 *
02406 *****
02407 *
02408 * ENTER *
02409 *
02410 *****
02411 *
02412 *
02413A EC41 4F EC41 A GETCR EQU #
02414A EC42 34 EC42 A GETCR1 EQU #
02415A EC42 97 34 A STA .KEY
02416A EC44 32 62 A LEAS REMOVE RETURN
02417 *
02418 * SEND MODIFIED FIELD
02419 *
02420 *
02421A EC46 96 34 A GETC EQU #
02422A EC48 87 D383 A LDA .KEY
02423A EC48 103E 54 A STA .COMAR+3
02424A EC4E 06 40 A STA
02425A EC50 A6 A4 A LDA #AMMO
02426A EC52 2A 02 EC56 GETC05 BPL GETC07
02427A EC54 33 A4 A LEAU
02428A EC56 E5 A0 A GETC07 BITB
02429A EC58 26 09 EC63 8NE GETC10
02430A EC5A 109C 56 A CMPLY GTBTMA
02431A EC5D 25 F1 EC5D BLO GETC05
02432 *
02433 * NO DATA
02434 *

```

```

02435A EC5F 7F D384 A CLR .COMAR+4
02436A EC62 38 RTS
02437 *
02438 EC63 A GETC10 EQU #
02439A EC63 31 3F A LEAY -1,Y SET THE FIRST OF MODIFY.
02440A EC65 E6 A4 A LDB
02441A EC67 C4 8F A GETC15 ANDB #FFF-AMMO
02442A EC69 E7 A0 A STB .Y CLEAR THE MODIFIED DATA TAG.
02443A EC68 E6 A4 A LDB
02444A EC6D 2A F8 EC67 BPL GETC15 MORE DATA IN FIELD.
02445A EC6F 31 C4 A LEAY
02446A EC71 30 C3 F830 A LEAX
02447A EC75 34 30 A PSHS
02448A EC77 9D A479 A JSR
02449A EC7A 26 30 A PULS
02450A EC7C CE D384 A LDU #.COMAR+4
02451A EC7F 86 03 A LDA #3
02452A EC81 A7 C0 A STA
02453A EC83 86 12 A LDA #12 SET FIELD
02454A EC85 A7 C0 A STA
02455A EC87 DC 58 A LDD
02456A EC89 1E 89 A EXG
02457A EC8B ED C1 A STD
02458A EC8D 31 21 A LEAY 1,Y PRE INC THE ATTR PTR.
02459 *
02460 EC8F A GETC20 EQU #
02461A EC9F A6 30 A LDA
02462A EC91 27 14 EC47 BEO GETC35
02463A EC93 1183 D400 A CMPLU #.COMAR+128
02464A EC97 25 09 ECA2 BLO GETC30
02465A EC99 8D E0F5 A JSR
02466A EC9C 7F D384 A CLR .COMAR+4
02467A EC9F CE D385 A LDU #.COMAR+5
02468 EC9A A GETC30 EQU #
02469A ECA2 A7 C0 A STA
02470A ECA4 7C D384 A INC .COMAR+4
02471A ECA7 E6 A0 A GETC35 LDB
02472A ECA9 2A E4 EC8F BPL GETC20
02473A ECAB 38 RTS
02474 *

```

```

02475 *
02476 * CONTROL KEY JUMP TABLE
02477 *
02478A ECAE E989 A GETJMP FDB GETNOP $00
02479A ECAE E983 A FDB GETNOP $01
02480A EC80 EBA4 A FDB CSLW $02
02481A EC82 EC3E A FDB GETCAN $03
02482A EC84 E989 A FDB GETNOP $04
02483A EC86 EAE0 A FDB ELKEY $05
02484A EC88 E856 A FDB CSRW $06
02485A EC8A E6A8 A FDB GETBEL $07
02486A EC8C E8DC A FDB GETBS $08
02487A EC8E E826 A FDB TAB $09
02488A EC90 E989 A FDB GETNOP $0A
02489A EC92 EA34 A FDB CSHOME $0B
02490A EC94 EAD2 A FDB GETCLR $0C
02491A EC96 EC41 A FDB GETCR $0D
02492A EC98 E989 A FDB GETNOP $0E
02493A EC9A E989 A FDB GETNOP $0F
02494A EC9C E989 A FDB GETNOP $10
02495A EC9E EB21 A FDB GETDUP $11
02496A EC98 EAE0 A FDB INSERT $12
02497A ECD2 E989 A FDB GETNOP $13
02498A ECD4 E989 A FDB GETNOP $14
02499A ECD6 E989 A FDB GETNOP $15
02500A ECD8 E989 A FDB GETNOP $16
02501A ECD4 E989 A FDB GETNOP $17
02502A ECD6 EC3E A FDB GETCAN $18
02503A ECD8 EAE0 A FDB GETBFO $19
02504A ECE0 EAB0 A FDB GETFFD $1A
02505A ECE2 E989 A FDB GETNOP $1B
02506A ECE4 EA14 A GETJCS FDB CSRL $1C
02507A ECE6 EA1F A FDB CSLL $1D
02508A ECE8 EA29 A FDB CSUL $1E
02509A ECEA EA37 A FDB CSOL $1F
02510 *
02511 * TTL GBADR - CONSOLE CODE -
02512 *****
02513 *
02514 * GET BUFFER ADDRESS *

```

```

02515 *
02516 *****
02517 *
02518 ECEC A GETBAD EQU
02519A ECEC DC 45 A LDA
02520A ECEE 1E 89 A EXG
02521A ECF0 FD D383 A STD
02522A ECF3 39 RTS
02523 *
02524 *****
02525 *
02526 * GET CHARACTER
02527 *
02528 *****
02529 *
02530 *
02531A ECF4 D0A0 A .$ORG SORG
02532A ECF4 D0A0 A CECMD SET
D0A1 A .$ORG DS
02533A ECF4 D0A1 A CBY1 EQU
D0A2 A .$ORG DS
02534A ECF4 D0A2 A CBY1 EQU
D0A3 A .$ORG DS
02535A ECF4 D0A3 A CBY2 EQU
D0A4 A .$ORG DS
02536A ECF4 D0A4 A CBY2 EQU
D0A5 A .$ORG DS
02537A ECF4 D0A5 A CBYTOP EQU
D0A6 A .$ORG DS
02538A ECF4 D0A6 A CPBAT EQU
D0A7 A .$ORG DS
02539A ECF4 D0A7 A CPBCNT EQU

```

```

02540A ECF4 D0A8 A .$ORG SET
D0A9 A CSAVE DS
D0AA A .$ORG SET
02541 *
02542A ECF4 86 A CGBLK2 FCB
A ECF5 4F A CGBLK1 STA
02543A ECF8 8D 29 ED23 BSR
02544A ECF8 CE D383 A LDU
02545A ECFD 6F C0 A CLR
02546A ECF8 8D 53 ED54 CGB100 BSR
02547A ED01 96 A0 A LDA
02548A ED03 27 04 ED06 BEO
02549A ED05 A6 A4 A LDA
02550A ED07 8D 06 ED0F BSR
02551A ED09 A6 84 A CGB120 LDA
02552A ED08 8D 02 ED0F BSR
02553A ED0D 20 F0 ECF4 BRA
02554 *
02555 ED0F A CGBDT EQU
02556A ED0F 1183 D400 A CMPLU
02557A ED13 25 08 ED1D BLO
02558A ED15 8D E0F5 A JSR
02559A ED18 CE D383 A LDU
02560A ED16 6F C0 A CLR
02561A ED1D 7C D383 A CGBDT5 STA
02562A ED20 A7 C0 A INC
02563A ED22 39 RTS
02564 *
02565 ED23 A CINIT EQU
02566A ED23 FC D383 A LDD
02567A ED26 8D E3D5 A JSR
02568A ED23 24 0A ED35 BCC
02569A ED26 97 A2 A STA
02570A ED2D 07 A1 A STB
02571A ED2F FC D385 A LDD
02572A ED32 8D E3D5 A JSR
02573A ED35 1024 F399 ED02 CBIN05 LBCC

```

```

02574A ED39 91 A2 A CMPLA
02575A ED3E 24 05 ED42 BHS
02576A ED3D 97 A2 A STA
02577A ED3F 66 D383 A STA
02578A ED42 97 A4 A CBIN50 STA
02579A ED44 D1 A1 A CMPE
02580A ED46 24 05 ED4D BHS
02581A ED48 D7 A1 A STB
02582A ED4A F6 D384 A STA
02583A ED4D 07 A3 A CBIN60 STA
02584A ED4F D6 A2 A LDB
02585A ED51 07 A5 A STB
02586A ED53 39 RTS
02587 *
02588 ED54 A CBLADR EQU
02589A ED54 DC A1 A LDA
02590A ED56 DD A8 A STD
02591A ED58 91 A3 A CMPLA
02592A ED5A 22 10 ED6C BHI
02593A ED5C DD E20C A JSR
02594A ED5F DC A1 A LDD
02595A ED61 5C A1 A INCB
02596A ED62 01 A4 A CMPE
02597A ED64 23 03 ED69 BLS
02598A ED66 4C A5 A INCA
02599A ED67 06 A1 A CBLAD7 STD
02600A ED69 DD A1 A CBLAD7 RTS
02601A ED68 39 62 A CBLEND LEAS
02602A ED6A 5C E27D A JNP
02603A ED6E 7E *****
02604 *
02605 *
02606 * PUT CHARACTER
02607 *
02608 *****
02609 *
02610 *
02611A ED71 86 A CGBLK2 FCB
A ED72 4F A CGBLK1 CLRA

```

02691A	EE03	34	80	A	OPR	#30	
02692A	EE05	20	04	EE0E	OPR	CURSTA	
02693A	EE07	96	1C	A	CURGO LDA	CURSPR	CLEAR THE CURSOR STOP.
02694A	EE09	84	7F	A	ANDA	#57F	
02695A	EE08	37	1C	A	CURSTA	STA	.CURSP
02696A	EE0D	39			RTS		
02698			0001	A	CSLCUP	EQU	#01
02699			0002	A	CSLOPD	EQU	#02
02700			0004	A	CSLTAB	EQU	#04
02701			0008	A	CSLPAG	EQU	#08
02702			0010	A	CSLSUS	EQU	#10
02703			0020	A	CSLDF	EQU	#20
02705			EE0E	A	INDCT	EQU	!
02706				!	PROGRAMMABLE	FUNCTION KEY'S STRING	
02707				!	INDICATOR.		
02708							
02709A	EE0E				SOFG	.BOXX1	
					SET	.BOXX1	
02710A	EE0E				DS	INDLP1,1	
					A INDLP1	EQU	.#0FG
					A .#0FG	SET	.#0FG+1
02711A	EE0E				DS	INDLP2,1	
					A INDLP2	EQU	.#0FG
					A .#0FG	SET	.#0FG+1
02712A	EE0E				DS	INDLP3,1	
					A INDLP3	EQU	.#0FG
					A .#0FG	SET	.#0FG+1
02713A	EE0E				DS	INDPF,1	
					A INDPF	EQU	.#0FG
					A .#0FG	SET	.#0FG+1
02714A	EE0E				DS	INDESV,1	
					A INDESV	EQU	.#0FG
					A .#0FG	SET	.#0FG+1
02715					!		
02716A	EE0E	96	38	A	LDA	..FS	
02717A	EE10	27	20	EE32	BEQ	INDIC3	NO DISPLAY DIRECT
02718A	EE12	86	0F	A	LDA	#15	NUMBER OF CHAR FOR DISPLAY
02719A	EE14	06	31	A	LDE	..CWD	IS THE CHARACTER SIZE SMALL?
02720A	EE16	27	02	EE1A	BEQ	INDIC2	YES.
02721A	EE18	86	07	A	LDA	#1	NO.

## FM-7 サブシステム・モニタ ソース・リスト

02722A	EE14	97	6C	A	INDOP	STA	INDOP		
02723A	EE1C	96	2F	A	LD	LD	LD	..ECL	
02724A	EE1E	97	2F	A	STA	INDBSV	ECL	SAVE	
02725A	EE20	9F	2F	A	CLF	..ECL	BACI	BLACI	
02726A	EE22	8E	D2C0	A	LD	..PFTBL			
02727A	EE25	96	43	A	LD	..DMAP+6			
02728A	EE27	4C			INCA				
02729A	EE28	8D	09	EE33	BSP	INDI			
02730A	EE2A	96	43	A	LD	..DMAP+6			
02731A	EE2C	8D	06	EE33	BSP	INDI			
02732A	EE3E	06	6D		LD	INDBSV			
02733A	EE39	07	2F	A	ST	..ECL			
02734A	EE32	39			INDIC9	PTS			
02735					+				
02736			EE33	A	INDI	EQU	+		
02737A	EE33	6D	E221	A	JSF	BVADL	VFAM	ADD. U	
02738A	EE36	1F	03	A	TF	D,U			
02739A	EE38	C6	06	A	LDB	#5			
02740A	EE3A	07	69	A	ST	INDLP1			
02741A	EE3D	8D	16	EE54	INDI3	BSP	PFDS		
02742A	EE3E	0A	69	A	DEC	INDLP1			
02743A	EE40	26	FA	EE3C	BNE	INDI3			
02744A	EE42	39			RTS				
02745					+				
02746A	EE43	4F			INDSPC	CLFA			
02747A	EE44	5F				CLF6			
02748					+				
02749			EE45	A	INDCHP	EQU	+		
02750A	EE45	34	56	A	PSHS	A,B,X,U			
02751A	EE47	36	C4	A	LEA	U			
02752A	EE49	6D	EE1D	A	JSF	CDS			
02753A	EE4C	EC	64	A	LDD	4,S			
02754A	EE4E	03	39	A	ADD	..VCU	VFAM	NEXT	
02755A	EE50	ED	64	A	STD	4,S			
02756A	EE52				RTSPUL	(A,B,X,U)			
A	EE52	25	D6	A	PULS	A,B,X,U,PC			
02757					+				
02758			EE54	A	PFDS	EQU	+		
02759A	EE54	34	10	A	PSHS	X	PF	PTF	SAVE

```

02763A EE58 2B 03 EE50 ENI YES.
02763A EE5A C6 00 LDB #50D M0X.
02763A EE5C
    A EE5C 8C A FFDSP3 LDB #8C CMPL #
02764A EE5D C6 0B A FFDSP3 LDB #50B
02765A EE5F 96 6C A LDB INDPF DISPLAY CHR
02766A EE61 97 6E A STA INDLP3 COUNT
02767A EE63 A6 80 A FFDSP5 LDA ,X+
02768A EE65 8D 0E EE45 BSF INDPCHP
02769A EE67 04 A 6E BSF INDL3
02770A EE69 26 F8 EE63 BNE FFDSP5
02771A EE6B 35 10 A PULS Y
02772A EE6D 30 88 10 A LEAX PFSZ,X NEXT PF
02773A EE70 20 D1 EE43 BRA INDSPC
02774
    F774 TTL GRAPHIC CODE - GRAPH
02775
    EE75 A CHRLN EQU #
02777A EE77 CL D380 A LOU ,.COMAP
02778A EE75 A6 46 A LDA 6,U
02779A EE77 E6 48 A LDB 8,U
02780A EE79 6D E305 A JSF CAXCF
02781A EE7C 24 07 EE85 BCF CTRLEP
02782A EE7E A6 4A A LDA 10,U
02783A EE80 E6 4C A LDB 12,U
02784A EE82 6D E305 A JSF CAXCF
02785A EE85 1024 F248 E0D2 CHRLP LBCC ERR61
02786A EE89 6F 45 A CLP 5,U
02787A EE8E 6F 47 A CLP 7,U
02788A EE8D 6F 48 A CLP 9,U
02789A EE8F 6F 4B A CLP 11,U
02790A EE91 86 FF A LDA #8FF
02791A EE93 67 D0A0 A STA LINCRR
02792A EE96 20 0E EEA6 BRA LINEDD
02793
    #
02794
    # LINE COMMAND
02795
    #
02796
    # ,.COMAP+3 COLOR CODE
02797
    # ,.COMAP+4 FUNCTION CODE
02798
    # ,.COMAP+5 X CO-ORDINATE OF P1
02799
    # ,.COMAP+7 Y CO-ORDINATE OF P1
02800
    #

```

```

02801      #
02802      #
02803      #
02804      #
02805      #
02806A EE98 FC EE98 A LINE EQU #
02807A EE9C 84 07 A LDD ,COMAR+3
02808A EE80 C1 05 A ANDA #807
02809A EE9F 24 23 A CMPE #5
02810A EEA1 D0 60 A STD ,GCL SET COLOR AND FUNCTION CODE.
02811A EEA3 BD E452 A JSR INITLN
02812A EEAE CE D385 A LINE00 LDU #CMOPPM+2
02813A EE4F 37 30 A PULU X,Y
02814A EEAE BD E442 A JSR GAXCH
02815A EEAE 9F 71 A STX ,LX1
02816A EEB0 109F 73 A STY ,LY1
02817A EEB3 37 30 A PULU X,Y
02818A EE85 BD E442 A JSR GAXCH
02819A EEB8 B6 D38D A LDA ,COMAR+13
02820A EEB6 26 0A EEC7 A BNE LINE10
02821      #
02822      #
02823      #
02824A EEB8 9F 75 A STX ,LX2
02825A EEBF 109F 77 A STY ,LY2
02826A EEC2 20 56 EF1A A JMP LINES
02827      #
02828A EEC4 7E E0DE A LINE90 BRF ERR64
02829      #
02830A EEC7 4A LINE10 DECA
02831A EEC8 26 2D EEF7 A BNE LINE20
02832      #
02833      #
02834      #
02835A EECA 9F 75 A STX ,LX2
02836A EEC8 BE D387 A LDX ,COMAR+7
02837A EECF 9F 77 A STX ,LY2
02838A EED1 8D 47 EF1A A BSR LINES X1,Y1-X2,Y1
02839A EED3 BE D385 A LDX ,COMAR+5
02840      #

```

02841A	EE08	9E	038B	A	LDX
02842A	EE08	9F	77	A	STX
02843A	EE0D	8D	38	EF1A	BSP
02844A	EE0F	BE	0389	A	LDX
02845A	EEEE	9F	71	A	STX
02846A	EEEE	9E	038B	A	LDX
02847A	EEEE	9F	73	A	STX
02848A	EE89	8D	2F	EF1A	BSP
02849A	EEEE	BE	0389	A	LDX
02850A	EEEE	9F	75	A	STX
02851A	EEFF	BE	0387	A	LDX
02852A	EEF3	9F	77	A	STX
02853A	EEF5	2D	23	EF1A	BRA
02854				‡	
02855		EEF7	A	LINE20	EQU
02856			‡		
02857			‡	PAINT	A
02858			‡		
02859A	EEF7	9E	75	A	STX
02860A	EEF9	109F	77	A	STX
02861A	EEFC	36	74	A	LDA
02862A	EEFE	06	78	A	LDB
02863A	EEF0	0D	74	A	SUBB
02864A	EF02	24	03	EF07	BHS
02865A	EF04	50			NEGB
02866A	EF05	36	78	A	LDA
02867A	EF07	37	74	A	LINE30
02868A	EF08	37	78	A	STA
02869A	EF0E	5C			INCB
02870A	EF0C	34	04	A	PSH
02871A	EF0E	8D	20	EF1A	LINE35
02872A	EF10	0C	74	A	INC
02873A	EF12	0C	78	A	INC
02874A	EF14	64	EA	A	DEC
02875A	EF16	26	F6	EF0E	BNE
02876A	EF18				RTSPUL
A	EF18	35	84	A	PULS
02877				‡	
02878				‡	

```

02880      #
02882      #
02883      #
02884      #
02885      #
02886      #
02887      #
02888      #
02889      #
02890      #
02891      #
02892      #
02893      #
02894      #
02895      #
02896      #
02897      #
02898      #
02899      #
02900      #
02901      #
02902      #
02903      #
02904      #
02905A EF1A 8E 0001 A LINES EQU
02906A EF10 CE FFFF A LDX
02907A EF20 CD 75 A LOD
02908A EF22 93 71 A SUBD
02909A EF24 28 04 A EF2A BHI
02910A EF26 3F A7 A STX
02911A EF28 20 08 EF30 A BFA
02912A EF2A 0F A7 A LINES1
02913A EF2C CD 71 A LDD
02914A EF2E 93 75 A SUBD
02915A EF30 C3 0001 A LINES2 ADDD
02916A EF32 0D A3 A STD
02917A EF35 0D A0 A STO
02918A EF37 CD 77 A LOD
02919A EF39 93 73 A SUEB
02920A EF3B 0D A3 A SUEB
02921A EF3D 0D A0 A STO
02922A EF3F CD 77 A LOD
02923A EF41 93 73 A SUEB
02924A EF43 0D A3 A SUEB
02925A EF45 0D A0 A STO
02926A EF47 CD 77 A LOD
02927A EF49 93 73 A SUEB
02928A EF4B 0D A3 A SUEB
02929A EF4D 0D A0 A STO
02930A EF4F CD 77 A LOD
02931A EF51 93 73 A SUEB
02932A EF53 0D A3 A SUEB
02933A EF55 0D A0 A STO
02934A EF57 CD 77 A LOD
02935A EF59 93 73 A SUEB
02936A EF5B 0D A3 A SUEB
02937A EF5D 0D A0 A STO
02938A EF5F CD 77 A LOD
02939A EF61 93 73 A SUEB
02940A EF63 0D A3 A SUEB
02941A EF65 0D A0 A STO
02942A EF67 CD 77 A LOD
02943A EF69 93 73 A SUEB
02944A EF6B 0D A3 A SUEB
02945A EF6D 0D A0 A STO
02946A EF6F CD 77 A LOD
02947A EF71 93 73 A SUEB
02948A EF73 0D A3 A SUEB
02949A EF75 0D A0 A STO
02950A EF77 CD 77 A LOD
02951A EF79 93 73 A SUEB
02952A EF7B 0D A3 A SUEB
02953A EF7D 0D A0 A STO
02954A EF7F CD 77 A LOD
02955A EF81 93 73 A SUEB
02956A EF83 0D A3 A SUEB
02957A EF85 0D A0 A STO
02958A EF87 CD 77 A LOD
02959A EF89 93 73 A SUEB
02960A EF8B 0D A3 A SUEB
02961A EF8D 0D A0 A STO
02962A EF8F CD 77 A LOD
02963A EF91 93 73 A SUEB
02964A EF93 0D A3 A SUEB
02965A EF95 0D A0 A STO
02966A EF97 CD 77 A LOD
02967A EF99 93 73 A SUEB
02968A EF9B 0D A3 A SUEB
02969A EF9D 0D A0 A STO
02970A EF9F CD 77 A LOD
02971A EFA1 93 73 A SUEB
02972A EFA3 0D A3 A SUEB
02973A EFA5 0D A0 A STO
02974A EFA7 CD 77 A LOD
02975A EFA9 93 73 A SUEB
02976A EFAB 0D A3 A SUEB
02977A EFAD 0D A0 A STO
02978A EFAF CD 77 A LOD
02979A EFB1 93 73 A SUEB
02980A EFB3 0D A3 A SUEB
02981A EFB5 0D A0 A STO
02982A EFB7 CD 77 A LOD
02983A EFB9 93 73 A SUEB
02984A EFBB 0D A3 A SUEB
02985A EFBD 0D A0 A STO
02986A EFBF CD 77 A LOD
02987A EFC1 93 73 A SUEB
02988A EFC3 0D A3 A SUEB
02989A EFC5 0D A0 A STO
02990A EFC7 CD 77 A LOD
02991A EFC9 93 73 A SUEB
02992A EFCB 0D A3 A SUEB
02993A EFD5 0D A0 A STO
02994A EFD7 CD 77 A LOD
02995A EFD9 93 73 A SUEB
02996A EFDB 0D A3 A SUEB
02997A EFD5 0D A0 A STO
02998A EFD7 CD 77 A LOD
02999A EFD9 93 73 A SUEB
03000A EFD5 0D A0 A STO
03001A EFD7 CD 77 A LOD
03002A EFD9 93 73 A SUEB
03003A EFD5 0D A0 A STO
03004A EFD7 CD 77 A LOD
03005A EFD9 93 73 A SUEB
03006A EFD5 0D A0 A STO
03007A EFD7 CD 77 A LOD
03008A EFD9 93 73 A SUEB
03009A EFD5 0D A0 A STO
03010A EFD7 CD 77 A LOD
03011A EFD9 93 73 A SUEB
03012A EFD5 0D A0 A STO
03013A EFD7 CD 77 A LOD
03014A EFD9 93 73 A SUEB
03015A EFD5 0D A0 A STO
03016A EFD7 CD 77 A LOD
03017A EFD9 93 73 A SUEB
03018A EFD5 0D A0 A STO
03019A EFD7 CD 77 A LOD
03020A EFD9 93 73 A SUEB
03021A EFD5 0D A0 A STO
03022A EFD7 CD 77 A LOD
03023A EFD9 93 73 A SUEB
03024A EFD5 0D A0 A STO
03025A EFD7 CD 77 A LOD
03026A EFD9 93 73 A SUEB
03027A EFD5 0D A0 A STO
03028A EFD7 CD 77 A LOD
03029A EFD9 93 73 A SUEB
03030A EFD5 0D A0 A STO
03031A EFD7 CD 77 A LOD
03032A EFD9 93 73 A SUEB
03033A EFD5 0D A0 A STO
03034A EFD7 CD 77 A LOD
03035A EFD9 93 73 A SUEB
03036A EFD5 0D A0 A STO
03037A EFD7 CD 77 A LOD
03038A EFD9 93 73 A SUEB
03039A EFD5 0D A0 A STO
03040A EFD7 CD 77 A LOD
03041A EFD9 93 73 A SUEB
03042A EFD5 0D A0 A STO
03043A EFD7 CD 77 A LOD
03044A EFD9 93 73 A SUEB
03045A EFD5 0D A0 A STO
03046A EFD7 CD 77 A LOD
03047A EFD9 93 73 A SUEB
03048A EFD5 0D A0 A STO
03049A EFD7 CD 77 A LOD
03050A EFD9 93 73 A SUEB
03051A EFD5 0D A0 A STO
03052A EFD7 CD 77 A LOD
03053A EFD9 93 73 A SUEB
03054A EFD5 0D A0 A STO
03055A EFD7 CD 77 A LOD
03056A EFD9 93 73 A SUEB
03057A EFD5 0D A0 A STO
03058A EFD7 CD 77 A LOD
03059A EFD9 93 73 A SUEB
03060A EFD5 0D A0 A STO
03061A EFD7 CD 77 A LOD
03062A EFD9 93 73 A SUEB
03063A EFD5 0D A0 A STO
03064A EFD7 CD 77 A LOD
03065A EFD9 93 73 A SUEB
03066A EFD5 0D A0 A STO
03067A EFD7 CD 77 A LOD
03068A EFD9 93 73 A SUEB
03069A EFD5 0D A0 A STO
03070A EFD7 CD 77 A LOD
03071A EFD9 93 73 A SUEB
03072A EFD5 0D A0 A STO
03073A EFD7 CD 77 A LOD
03074A EFD9 93 73 A SUEB
03075A EFD5 0D A0 A STO
03076A EFD7 CD 77 A LOD
03077A EFD9 93 73 A SUEB
03078A EFD5 0D A0 A STO
03079A EFD7 CD 77 A LOD
03080A EFD9 93 73 A SUEB
03081A EFD5 0D A0 A STO
03082A EFD7 CD 77 A LOD
03083A EFD9 93 73 A SUEB
03084A EFD5 0D A0 A STO
03085A EFD7 CD 77 A LOD
03086A EFD9 93 73 A SUEB
03087A EFD5 0D A0 A STO
03088A EFD7 CD 77 A LOD
03089A EFD9 93 73 A SUEB
03090A EFD5 0D A0 A STO
03091A EFD7 CD 77 A LOD
03092A EFD9 93 73 A SUEB
03093A EFD5 0D A0 A STO
03094A EFD7 CD 77 A LOD
03095A EFD9 93 73 A SUEB
03096A EFD5 0D A0 A STO
03097A EFD7 CD 77 A LOD
03098A EFD9 93 73 A SUEB
03099A EFD5 0D A0 A STO
03100A EFD7 CD 77 A LOD
03101A EFD9 93 73 A SUEB
03102A EFD5 0D A0 A STO
03103A EFD7 CD 77 A LOD
03104A EFD9 93 73 A SUEB
03105A EFD5 0D A0 A STO
03106A EFD7 CD 77 A LOD
03107A EFD9 93 73 A SUEB
03108A EFD5 0D A0 A STO
03109A EFD7 CD 77 A LOD
03110A EFD9 93 73 A SUEB
03111A EFD5 0D A0 A STO
03112A EFD7 CD 77 A LOD
03113A EFD9 93 73 A SUEB
03114A EFD5 0D A0 A STO
03115A EFD7 CD 77 A LOD
03116A EFD9 93 73 A SUEB
03117A EFD5 0D A0 A STO
03118A EFD7 CD 77 A LOD
03119A EFD9 93 73 A SUEB
03120A EFD5 0D A0 A STO
03121A EFD7 CD 77 A LOD
03122A EFD9 93 73 A SUEB
03123A EFD
```

02921A	EF3D	3F	A9	A	STX
02922A	EF30	20	06	EF47	BRA
02923A	EF41	0F	A9	A	LINES3
02924A	EF43	0C	73	A	STD
02925A	EF45	38	77	A	SUB0
02926A	EF47	C3	0001	A	LINES4
02927A	EF4A	DD	A5	A	ADD
02928A					STD
02929A	EF46	36	A0	A	#
02930A	EF4E	26	0E	EF5E	A
02931A	EF50	9E	A3	A	LDA
02932A	EF52	30	1F	A	BNE
02933A	EF54	27	74	EFCA	L
02934A	EF56	9E	A5	A	LEAX
02935A	EF58	30	1F	A	LDD
02936A	EF5A	102F	0088	EF65	A
02937					LEAQ
02938A	EF5E	CC	0000	A	LNCDS
02939A	EF61	DD	A6	A	STD
02940A	EF63	DD	71	A	LDD
02941A	EF65	DD	64	A	STD
02942A	EF67	DD	73	A	LDD
02943A	EF69	DD	62	A	STD
02944					
02945					#
02946A	EF6B			A	LINES5
02947A	EF6D			A	EQU
02948A	EF6E	0C	AB	A	JSC
02949A	EF70	03	A5	A	LDD
02950A	EF72	DD	AB	A	ADD
02951					STD
02952A	EF74	93	A3	A	#
02953A	EF76	23	0F	EF87	A
02954A	EF78	DD	AB	A	LINES6
02955A	EF7A	DE	63	A	STD
02956A	EF7C	DE	AA	A	LDB
02957A	EF7E	7D	63	A	ADD
02958					STB
02959					#
02960A	EF80	BD	FEA0	A	LDD
02961					LNPI
02962					#
02963					SUBD
02964					LINDX
02965					#
02966A	EF80	BD	FEA0	A	LDD

.COMAR+11 .LY2 LINES X1,Y1-X1,Y2 .COMAR+8 .LX1 .COMAR+11 .LY1 LINES X2,Y2-X1,Y2 .COMAR+9 .LX2 .COMAR+7 .LY2 LINES X2,Y2-X2,Y1	02961A EF83 DC AB A LDD LNPY 02962A EF85 20 ED EF74 BRA LINES6 END1 02963 02964 EF87 A LINES7 EQU # 02965A EF87 26 08 EF91 BNE LINES8 PY .NE. DX THEN GOTO LINES8 02966A EF88 00 AB A STD LNPY PY=01 02967A EF88 06 83 A .GV+1 02968A EF88 06 AA A ADD8 LNUY+1 02969A EF8F D7 63 A STB .GV+1 02970A EF91 DC 64 A LINES8 LDD .GX 02971A EF93 D3 A7 A ADDD LNUX .GX 02972A EF95 00 64 A STD .GX 02973A EF97 9E AD A STD LNLN GX=GX+UX1 02974A EF99 30 1F A LEAX -1,X 02975A EF98 9F AD A STX LNLN 02976A EF9D 26 CC EF6B BNE LINES5 02977A EF9F 39 LINES9 RTS 02978 02979A EFA0 96 A0 A LNPOIN LDA LNCHP 02980A EFA2 27 23 EFC7 BEQ LNDOT 02981A EFA4 08 65 A LDB .GX+1 02982A EFA6 96 63 A LDA .GV+1 02983A EFA8 34 06 A PSHS D 02984A EFAA ED E20C A JSR .BADR 31/7/7 02985A EFAD 35 06 A PULS D 02986A EFAD 60 E225 A JSR EVAD 02987A EFB2 34 06 A PSHS D 02988A EFB4 65 D384 A LDA .COMAR+4 02989A EFB7 A7 84 A STA .Y 02990A EFB9 E6 A4 A LDB .Y 02991A EFB6 C4 80 A AND8 #580 02992A EFB8 FA D383 A OR8 .COMAR+3 02993A EFC0 E7 A4 A STE .Y 02994A EFC2 35 10 A PULS X 02995A EFC4 7E E510 A JMP CDSP 02996A EFC7 7E E48F A LNDOT GPDSP 02998 02999 03000A EFCA 9E 71 A LINEY LDX .LX1 03001A EFCC 103E 73 A LDY .LY1 ASSUME POINT-1 COORDINATES.	
RECTANGLE .LX2 .LY2 .LY1+1 .LY2+1 .LY1+1 LINE30 .LY2+1 .LY1+1 .LY2+1 B LINES .LY1+1 .LY2+1 S LINE35 B B,PC	03002A EFCF 96 A9 A LDA LNUY 03003A EFD1 2A 03 EFD6 BPL LLINEY2 03004A EFD3 103E 77 A LDY .LY2 03005A EFD6 ED E3DC A LINEY2 JSR GADDP 03006A EFD8 30 C4 A LINEY4 LEAX SAVE IT TO X. 03007A EFD8 BD E437 A JSR GPDSPR 03008A EFD8 33 88 50 A LEAU GXES2,X 03009A EFE1 0A A6 A DEC LNDY+1 03010A EFE3 26 F4 EFD3 BNE LLINEY4 03011A EFE5 39 LLINEY RTS 03012 03013 03014 03015A EFE6 9E 71 A LINEX LDX .LX1 03016A EFE8 DE 75 A LDU .LX2 03017A EFEA 96 A7 A LDA LNUX 03018A EFE2 2A 02 EFD0 BPL LLINEX2 03019A EFE2 1E 13 A EFG LNUX 03020A EFE3 9F A7 A LINEX2 STX LNUY 03021A EFE3 DF A9 A STU LNUY 03022A EFE4 96 A8 A LDA LNUX+1 03023A EFE6 84 07 A ANDA #507 03024A EFE8 8E E392 A LDX #BAS 03025A EFE8 E6 86 A LDB A,X 03026A EFE8 98 LSLB 03027A EFFE 5A DECB 03028A EFFF 96 AA A LDA LNUY+1 03029A F001 84 07 A ANDA #507 03030A F003 A6 86 A LDA A,X 03031A F005 4A DECA 03032A F006 43 COMA 03033A F007 34 06 A PSHS 03034A F008 DC A7 A LDD LNUX 03035A F008 BD E3FB A JSR LSRD3 03036A F00E 34 04 A PSHS B 03037A F010 DC A9 A LDD LNUY 03038A F012 BD E3FB A JSR LSRD3 03039A F015 E0 E0 A SUBB LNUX 03040A F017 26 07 F020 BNE LLINEX3 03041A F018 E8 E0 A LDB .S+ IS IT A NEGATIVE DIRECTION? NO. YES, THE START POINT IS THE POINT2. GET THE VRAM ADDRESS OF START POINT. SAVE IT TO X. DISPLAY ADOT.	
STRAIGHT LINE CO-ORDINATE OF POINT1 CO-ORDINATE OF POINT11 CO-ORDINATE OF POINT2 CO-ORDINATE OF POINT2 .WOPK LNCHR+1 LNBPY+2 LNDX+2 LNDY+2 LNUX+2 LNUY+2 LNPY+2 LNLN+2 # #1 1 UNIT #-1 -1 UNIT .LX2 .LX1 DX=X2-X11 LINES1 IF DX=0 LNUX THEN UX=11 LINES2 LNUX ELSE DO1 UX=-11 .LX1 DX=X1-X21 .LX2 END1 #1 DX=DX+11 LNDX LNLN .LY2 .LY1 LINES3 IF DY=0 DY=Y2-Y11 IF DY>0	03042A F018 E4 E0 A AND8 .S+ 03043A F01D 7E F05A A JMP LNRSID 03045A F020 D7 AD A LINEX3 STB LNLN 03046A F022 E6 E0 A LDB .S+ 03047A F024 C1 FF A CMPPB #5FF 03048A F026 27 05 F02D BEQ LLINEX4 03049A F028 BD F05A A JSR LNRSID 03050A F028 0A AD A DEC LNLN 03051A F02D E6 E0 A LINEX4 LDB .S+ 03052A F02F 28 0D F03E BMI LLINEX5 03053A F031 BD F0BD A JSR LNLSID 03054A F038 9E A7 A LDX LNUX 03055A F038 30 08 A LEAX .GX 03056A F038 9F A7 A STX LNUX 03057A F03A 0A AD A DEC LNLN 03058A F03C 2B A7 EFE5 BMI LLINEX 03059 F03E A LINEX5 ECU # 03060A F03E 103E 73 A LDY .LY1 03061A F041 5E A7 A LDX LNUX 03062A F043 BD E3DC A JSR GADDP 03063A F046 D6 AD A LDB .GV+1 03064A F048 9D 8C A JSR BLULIN 03065A F04A 33 C9 4000 A LEAU .VR,U 03066A F04E D5 AD A LDB LNLN 03067A F050 9D 8F A JSR REDLIN 03068A F052 33 C9 4000 A LEAU .VG-.VR,U 03069A F056 D6 AD A LDB LNLN 03070A F058 0E 92 A JMP GRELIN 03071 03072A F05A 9E A9 A LNRSID LDX LNUY 03073A F05C A F05C A FCB #8C 03074A F05D 9E A7 A LNLSID LDX LNUX 03075A F05F 103E 73 A LDY .LY1 03076A F062 34 04 A PSHS B 03077A F064 BD E3DC A JSR GADDP 03078A F067 30 C4 A LEAX .U 03079A F069 31 E4 A LEAY .S LOAD THE RIGHT X COORDINATE. CMPP # LOAD THE LEFT X COORDINATE. LOAD THE Y COORDINATE. SAVE BIT MASK. GET VRAM ADDR. TFR U TO X. Y POINT TO BIT MASK.	
LNUN LINES4 LNUY .LY1 .LY2 END1 #1 LNDY LNCHR LNCDSP LNDX -1,X LINEY -1,X LINEX #0 LNPY PY=01 .LX1 .GX .LY1 .GY GX=X11 GY=Y11 * LNPOIN CALL GPDSP1 LNPY LNPY PY=PY+DY1 LNDX DO WHILE (PY>DX) 1 LNPY .GY+1 .GY+1 LNPIN CALL GPDSP1	03042A F018 E4 E0 A AND8 .S+ 03043A F01D 7E F05A A JMP LNRSID 03045A F020 D7 AD A LINEX3 STB LNLN 03046A F022 E6 E0 A LDB .S+ 03047A F024 C1 FF A CMPPB #5FF 03048A F026 27 05 F02D BEQ LLINEX4 03049A F028 BD F05A A JSR LNRSID 03050A F028 0A AD A DEC LNLN 03051A F02D E6 E0 A LINEX4 LDB .S+ 03052A F02F 28 0D F03E BMI LLINEX5 03053A F031 BD F0BD A JSR LNLSID 03054A F038 9E A7 A LDX LNUX 03055A F038 30 08 A LEAX .GX 03056A F038 9F A7 A STX LNUX 03057A F03A 0A AD A DEC LNLN 03058A F03C 2B A7 EFE5 BMI LLINEX 03059 F03E A LINEX5 ECU # 03060A F03E 103E 73 A LDY .LY1 03061A F041 5E A7 A LDX LNUX 03062A F043 BD E3DC A JSR GADDP 03063A F046 D6 AD A LDB .GV+1 03064A F048 9D 8C A JSR BLULIN 03065A F04A 33 C9 4000 A LEAU .VR,U 03066A F04E D5 AD A LDB LNLN 03067A F050 9D 8F A JSR REDLIN 03068A F052 33 C9 4000 A LEAU .VG-.VR,U 03069A F056 D6 AD A LDB LNLN 03070A F058 0E 92 A JMP GRELIN 03071 03072A F05A 9E A9 A LNRSID LDX LNUY 03073A F05C A F05C A FCB #8C 03074A F05D 9E A7 A LNLSID LDX LNUX 03075A F05F 103E 73 A LDY .LY1 03076A F062 34 04 A PSHS B 03077A F064 BD E3DC A JSR GADDP 03078A F067 30 C4 A LEAX .U 03079A F069 31 E4 A LEAY .S LOGICAL AND THE BOTH SIDE BIT MASK. SET BIT IMG. AND RETURN SAVE IS THE RIGHT SIDE FULL BYTE? YES. NO, PUT BYTE WITH MASK. IS THE LEFT SIDE FULL BYTE? YES. NO, PUT BYTE WITH MASK. LEFT SIDE IS POINTED NEXT. LNUX LNLN NO BYTE REMAIN. GO FINISH. LOAD THE Y COORDINATE. LOAD THE LEFT X. GET LOWEST DOT ADDRESS.	



# FM-7 サブシステム・モニタ ソース・リスト

```

03080A F06B 5F          CLR8
03081A F06C 90          JSR  BLUSID  BLUE
03082A F06E 30          LEAX  ..VR,U  X PTR RED VRAM
03083A F072 5F          CLR8
03084A F073 90          JSR  REDSID  RED
03085A F075 30          LEAX  ..VG,U  X PTR GREEN VRAM
03086A F079 5F          CLR8
03087A F07A 90          JSR  GRESID  GREEN
03088A F07C          RTSPUL  E
      A F07C 35          PULS  E,PC
03090A F07E          F631  A LNXFNC FDB LNXOP,LNXAND PSET
      A F080          F647  A
03091A F082          F631  A FDB LNXOP,LNXAND PRESET
      A F084          F647  A
03092A F086          F631  A FDB LNXOP,LNXAND OR
      A F088          F657  A
03093A F08A          F657  A FDB LNXNOP,LNXAND AND
      A F08C          F647  A
03094A F08E          F65E  A FDB LNXEOP,LNXNOP EOP
      A F090          F657  A

03096
03097A F092 4F          LNAND CLFA
03098A F093          8C  A FCE SKIP2
      A F093          FF  A LNOR LDA  #8C  CMPX #
03099A F094 86          FF  A LNOR VACON
03100A F096          8C  A LNOR VACON
      A F096 B5          D409 A BITA VACC
03101A F099 12          NOP
03102A F09A A7          C5  A LNORDD STA  B,U
03103A F09C 5A          FB  F09A LNORDD DEC8
03104A F09D 2A          FB  F09A LNORDD BPL
03105A F09F          D409 A LNORDD VACOFF
      A F09F B7          D409 A LNORDD STA  VACC
03106A F0A2 39          LNORDD RTS
03107
03108A F0A3          D409 A LNORDD VACON
      A F0A3 B5          D409 A LNORDD BITA VACC
03109A F0A6 12          NOP
03110A F0A7 63          C5  A LNORDD COM  B,U
03111A F0A9 5A          COM  DEC8

```

```

03112A F0AA 2A          FB  F0A7 BPL LNEORD
03113A F0AC          D409 A STA VACOFF
      A F0AC B7          D409 A STA VACC
03114A F0AF 39          FB  F0A7 RTS
03115
03116A F0B0          F094 A LNUNFC FDB LNOR,LNAND
      A F0B2          F092 A
03117A F0B4          F094 A FDB LNOR,LNAND
      A F0B6          F092 A
03118A F0B8          F094 A FDB LNOR,LNNOP
      A F0BA          F0A2 A
03119A F0BC          F0A2 A FDB LNNOP,LNAND
      A F0BE          F092 A
03120A F0C0          F0A3 A FDB LNEOR,LNNOP
      A F0C2          F0A2 A

03121
03122          *** END OF SUBROUTINE "LINES"
03123          *
03124          *
03125          *
03126          * CHAIN COMMAND
03127          *
03128          * .COMAR+3 COLOR CODE
03129          * .COMAR+4 FUNCTION CODE
03130          * .COMAR+5 NUMBER OF POINT
03131          * .COMAR+6 CHAINED POINTS
03132
03133          *
03134A F0C4 FC          F0C4 A CHAIN EQU 1
03135A F0C7 C1          D383 A LD  CNDPRM A=COLOR,B=FUNCTION
      A F0C7 C1          05  A CMPE #5
03136A F0C9 24          32  FOFD BHS CHAIN7 FUNCTION CODE ERROR
03137A F0CB DD          60  A ST  .GCL
03138A F0CD BD          E452 A JSR  INITLN
03139A F0DD CE          D385 A LD  CNDPRM+2
03140A F0E3 A6          00  A LDA  .U+
03141A F0E5 81          01  A CMPA #1
03142A F0E7 23          27  F100 BLS CHAIN8 NUMBER OF POINTS ERROR
03143A F0E9 81          1E  A CMPA #30
03144A F0EB 22          23  F100 BHI CHAIN8 NUMBER OF POINTS ERROR
03145          *

```

```

03146A F0DD 4A          DECA
03147A F0DE 37          30  A PULU X,Y
03148A F0E0 BD          E442 A JSR  GAXCHK
03149A F0E3 9F          71  A CHAIN2 STX  .LX1
03150A F0E5 109F       73  A STY  .LY1
03151A F0E8 37          30  A PULU X,Y
03152A F0EA BD          E442 A JSR  GAXCHK
03153A F0ED 9F          75  A STX  .LX2
03154A F0EF 109F       77  A STY  .LY2
03155A F0F2 34          72  A PSHS X,Y,U,A
03156A F0F4 BD          EF1A A JSR  LINES
03157A F0F7 35          72  A PULS X,Y,U,A
03158A F0F9 4A          DECA
03159A F0FB 26          E7  F0E3 BNE  CHAIN2
03160A F0FC 39          RTS

03161
03162A F0FD 7E          E0DB A CHAIN7 JMP  ERR64
03163A F100 7E          E0DE A CHAIN8 JMP  ERR65
03164          *
03165          *
03166          * END OF "CHAIN" COMMAND
03167          *
03168          *
03169          *
03170          * SYMBOL COMMAND
03171          * .COMAR+3 COLOR CODE
03172          * .COMAR+4 FUNCTION CODE
03173          * .COMAR+5 ANGLE CODE
03174          * .COMAR+6 MUL OF WIDTH
03175          * .COMAR+7 MUL OF HEIGHT
03176          * .COMAR+8 X COORDINATES
03177          * .COMAR+10 Y COORDINATES
03178          * .COMAR+12 NUMBER OF CHARACTER
03179          * .COMAR+13 CHARACTER STRING
03180          *
03181          *
03182          DOA0  A WAX  EQU  .WORK PTR OF WK W AX
03183          DOA2  A HAX  EQU  .WORK+2 PTR OF WK H AX
03184          DOA4  A USV  EQU  .WORK+4 W DOT STEP
03185          DOA6  A HSV  EQU  .WORK+6 H DOT STEP
03186          DOA8  A WCV  EQU  .WORK+8 W CHR DOT STEP

```

```

03187          DOA8  A HCSV
03188          DOAC  A NOTF
03189          DOAD  A CPTR
03190          DOAF  A WOP
03191          DOB1  A HOP
03192          DOB3  A FPRK
03193          DOB4  A FPTR
03194          DOB6  A LW
03195          DOB7  A LH
03196          DOB8  A SYMFT
03197          *
03198          *
03199          *
03200A F103 FC          Q383 A SYMBOL
03201A F106 0F          AC  A
03202A F108 C1          05  A
03203A F10A 22          F1  FOFD
03204A F10C 26          03  F111
03205A F10E 03          AC  A
03206A F110 9F          8F  A
03207A F111 BD          E468 A SYMB00
03208A F114 B5          D385 A
03209A F117 44          CC  A
03210A F118 CC          D064 A
03211A F11B 8E          D062 A
03212A F11E 10BE       D38A A
03213A F122 CE          D38A A
03214A F125 24          04  F12B
03215A F127 1E          01  A
03216A F129 1E          23  A
03217A F12B 34          76  A SYMB01
03218          *
03219          * 0,S
03220          * 2,S
03221          * 4,S
03222          * 6,S
03223          *
03224A F12D CE          D385 A
03225A F130 4F          4F  A

```

```

03226A F131 E6          41  A
03227A F133 27          04  F138
03228A F135 DD          A8  A
03229A F137 E6          42  A
03230A F139 10C7       EFAD E0EA SYMB04
03231A F13B DD          A4  A
03232A F13F E5          C4  A
03233A F141 C4          03  A
03234A F143 C1          02  A
03235A F145 25          02  F149
03236A F147 C8          01  A
03237A F149 34          04  A
03238A F14B 5F          5F  A
03239A F14D 64          E4  A
03240A F14E 24          06  F156
03241A F150 33          A8  A
03242A F152 DD          A8  A
03243A F154 C6          FE  A
03244A F156 5C          5C  A SYMB05
03245A F157 DD          A4  A
03246A F159 4F          4F  A
03247A F15A 5F          5F  A
03248A F15B 64          E0  A
03249A F15D 24          06  F165
03250A F15F 33          AA  A
03251A F161 DD          AA  A
03252A F163 C6          FE  A
03253A F165 5C          5C  A SYMB06
03254A F166 DD          A6  A
03255A F168 8E          D38D A
03256A F16B 9F          AD  A
03257          F16D A SYMB10
03258A F16D B6          D38C A
03259A F170 26          02  F174
03260A F172          A F172 35  F6  A
03261A F174 A6          9F DOAD A SYMB13
03262A F178 BD          E387 A
03263A F17B 10BE       D0B8 A
03264A F17F C6          06  A
03265A F181 A6          80  A SYMB15

```

```

03266A F183 0D          AC  A
03267A F185 27          01  F188
03268A F187 43          A0  A
03269A F188 A7          A0  A
03270A F18A 5A          5A  A
03271A F18B 26          F4  F181
03272A F18D FC          D388 A
03273A F190 DD          64  A
03274A F192 FC          D38A A
03275A F195 DD          62  A
03276A F197 DD          E383 A
03277A F19A 25          02  F19E
03278A F19C          A F19C 35  F6  A
03279A F19E 58          80  A SYMB17
03280A F1A0 97          B3  A
03281          F1A2 A SYMB20
03282A F1A2 AE          F8 06 A
03283A F1A5 AF          F8 02 A
03284A F1A8 8E          D0B8 A
03285          F1AB A SYMB30
03286A F1AB 9F          B4  A
03287A F1AD BD          E383 A
03288A F1B0 24          08  F1BA
03289A F1B2 96          B3  A
03290A F1B4 A5          9F D0B4 A
03291A F1B6 26          0A  F1C4
03292          F1BA A SYMB33
03293A F1BA DC          AA  A
03294A F1BC AE          62  A
03295A F1BE E3          84  A
03296A F1C0 ED          84  F1F8
03297A F1C2 20          34  F1F8
03298          *
03299          *
03300A F1C4 B6          D387 A
03301A F1C7 97          B7  A
03302          F1C9 A SYMB40
03303A F1C9 B6          D386 A
03304A F1CC 97          B6  A

```

```

EQU .WORK+10 H CHR DOT STEP
EQU .WORK+12 FONT NOT FLAG
EQU .WORK+13 CHARACTER PTR
EQU .WORK+15 W ORG
EQU .WORK+17 H ORG
EQU .WORK+19 FONT TEST MASK
EQU .WORK+20 FONT PTR
EQU .WORK+22 LOOP W
EQU .WORK+23 LOOP H
EQU .WORK+24 TEMP FONT AREA

```

```

EQU #CDPRM ENTRY
CLR NOTF
CMPB #5 NOT?
BHI CHAIN7
BNE SYMB00 NO.
COM NOTF FUNCTION IS PSET
JSR GPODSIR
LDA .COMAR+5 ANGLE
LSRA
LDD #.GX
LDX #.GY
LDY #.COMAR+8
LDJ #.COMAR+10
ECC SYMB01
EXG D,X
EXG Y,U
PSHS U,Y,X,D

```

COORDINATES PTR OF WIDTH AXIS  
 COORDINATES PTR OF HEIGHT AXIS  
 ORIGINAL COORDINATES PTR OF WIDTH AXIS  
 ORIGINAL COORDINATES PTR OF HEIGHT AXIS

```

LDU #.COMAR+5
CLRA
LDE 1,U MUL OF WIDTH

```

```

03305 F1CE A SYMB50 EQU #
03306A F1CE BD E4F A JSR GPOSP ***DISPLAY***
03307A F1D1 AE E4 A LDX .S
03308A F1D3 EC 34 A LDD .X
03309A F1D5 D3 A4 A ADDD WSV
03310A F1D7 ED 64 A STD .X
03311A F1D3 BD E3B3 A JSR GAXC
03312A F1D0 24 04 F1E2 BCC SYMB55 OVER
03313A F1D0 0A B6 A DEC LV
03314A F1E0 26 EC F1CE BNE SYMB50
03315 F1E2 A SYMB55 EQU #
03316A F1E2 EC F8 04 A LDD [4,S]
03317A F1E2 ED F4 A STD [5,S]
03318A F1E7 AE 62 A LDX .S
03319A F1E9 EC 34 A LDD .X
03320A F1E8 D3 A6 A ADDD HSV
03321A F1ED ED 34 A STD .X
03322A F1EF BD E3B3 A JSR GAXC
03323A F1F2 24 0D F201 BCC SYMB70
03324A F1F4 0A 67 A DEC LN
03325A F1F8 26 D1 F1C3 BNE SYMB40
03326 F1F8 A SYMB60 EQU #
03327A F1F8 9E B4 A LDX FPTR
03328A F1FA 30 01 A LEAX 1,X PTR NEXT FONT DOT
03329A F1FC 8C D0C0 A CMPX #SYMFNT+CFH END ?
03330A F1FF 26 AA F1AB BNE SYMB30 NO
03331 F201 A SYMB70 EQU #
03332A F201 AE 64 A LDX 4,S
03333A F203 EC 34 A LDD .X
03334A F205 D3 A8 A ADDD WCSV
03335A F207 ED 34 A STD .X
03336A F209 ED F4 A STD [5,S]
03337A F20B 04 B3 A LSR FMKX SHIFT RIGHT MARK
03338A F20D 26 32 F1A2 BNE SYMB20
03339A F20F 7A D3C A DEC .COMAR+12 ONE CHR END
03340A F212 9E AD A LDX CPTR PTR NEXT CHR
03341A F214 30 01 A LEAX 1,X
03342A F216 9F AD A STX CPTR
03343A F218 7E F16D A JMP SYMB10
03344 #

```

```

REQ SYMB04
STD WCSV
LDB 2,U MUL OF HEIGHT
LBEQ ERR99 PARM ERROR
STD HCSV
LDE #U ANGLE CODE
ANDB #503
CMPB #2
BLO #+4
EORB #501
PSHS B
CLRB
LSR .S
ECC SYMB05
SUBD WCSV
STD WCSV
LDB #5FE
INCB
STD WSV
CLRB
LSR .S+
ECC SYMB06
SUBD HCSV
STD HCSV
LDB #5FE
INCB
STD HSV
LDX #.COMAR+13
STX CPTR VHAR PTR
EQU # START DISPLAY
LDA .COMAR+12 CHAR END?
BNE SYMB13 NO, GO DO MORE.
RTSPUL (U,Y,X,D)
PULS (U,Y,X,D,PC)
LDA [CPTR] CHR LOAD
JSR CFADR FONT ORIGINAL
LDY #SYMFNT FONT STORE ADDR
LDB #CFH NUMBER OF FONT BYTE
LDA 0,X+

```

```

03345 *****
03346 #
03347 #
03348 # CHANGE COMMAND
03349 #
03350 #
03351 # .COMAR+3 X1
03352 # .COMAR+5 Y1
03353 # .COMAR+7 X2
03354 # .COMAR+9 Y2
03355 # .COMAR+11 NUMBER OF COLOR
03356 # .COMAR+12 COLOR
03357 #
03358 #
03359 D0A0 A CHGCL EQU .WORK
03360 D0A8 A CHG1 EQU CHGCL+8
03361 D0A9 A CHGJ EQU CHG1+1
03362 D0AA A CHG1 EQU CHGJ+1
03363 #
03364 #
03365A F218 BD F21E A CHANGE EQU #
03366A F21E D7 AA A JSR BOXINV
03367A F220 8E D0A0 A STD CHG1
03368A F223 86 07 A LDA #CHGCL
03369A F225 A7 36 A LDA #7
03370A F227 4A # A CHANG0 STA A,X
03371A F228 2A FB F225 BPL CHANG0 CHANGE TABLE INIT VALUE
03372A F22A F6 D3B8 A LDE .COMAR+11
03373A F22D 27 51 F280 BEQ CHANGX
03374A F22F C1 08 A CMPB #8
03375A F231 22 4D F280 BHI CHANGX
03376A F233 108E D3B8 A LDY #.COMAR+12
03377A F237 D7 A8 A STE CHG1
03378A F239 EC A1 A CHANG1 LDD 0,Y++
03379A F23B 84 07 A ANDA #7
03380A F23D C4 07 A ANDB #7
03381A F23F E7 36 A STE A,X
03382A F241 0A A8 A DEC CHG1
03383A F243 26 F4 F239 BNE CHANG1 CHANGE TABLE MADE
03384 #
03385A F245 0F 61 A CLR .GF FUNCTION IS "SET"

```

```

TST NOTF
BEQ #+3
COMA NOT FUNCTION
STA 0,Y+ STORE
DECB
BNE SYMB15 REMAIN
LDD .COMAR+8
STD .GX
LDD .COMAR+10
STD .GY
JSR GAXC CHECK AXIS
BCS SYMB17
RTSPUL (U,Y,X,D)
PULS (U,Y,X,D,PC)
LDA #80
STA FMKX TEST MARK INIT
EQU #
LDX [6,S]
STX [2,S]
LDX #SYMFNT FONT ADDR INIT
EQU #
STX FPTR SET FONT PTR
JDX GAXC
ECC SYMB33
LDA FMKX
BITA [FPTR] DOT ON?
BNE SYMB35 ON
EQU #
LDD HCSV CLEAR
LDX 2,S
ADD #X
STD #X
BRA SYMB60
EQU #
LDA .COMAR+7
STA LH HEIGHT DOT
EQU #
LDA .COMAR+6 WIDTH DOT
STA LW

```

```

03386 #
03387A F247 9E F247 A CHANG4 EQU #
03388A F249 109E 6B A LDX .BOXX1
03389A F24C 36 AA A LDY .BOXY1
03390A F24E 4C A LDA CHGK
03391A F24F 37 A3 A INCA
03392A F251 BD E3DC A STA CHGJ
03393A F254 8E D0A0 A GADDR
03394 F257 A CHANG5 EQU #
03395A F257 BD E39A A LDX #CHGCL
03396A F25A A6 86 A JSR GPSETR
03397A F25C 37 60 A LDA A,X
03398A F25E 34 54 A STA .GCL
03399A F260 BD E469 A PSHS X,U,B
03400A F263 35 54 A JSR GPDSPI
03401A F265 34 54 A PULS X,U,B
03402A F267 BD E437 A JSR GPDSPI
03403A F26A 35 54 A PULS X,U,B
03404A F26C 33 C8 50 A LEAU GB5Z,U
03405A F26F 0A A9 A DEC CHGJ
03406A F271 26 E4 F257 BNE CHANG5
03407 #
03408A F273 9E 89 A LDX .BOXX2
03409A F275 9C 8D A LDY .BOXY2
03410A F277 27 06 F27F BEQ CHANG9
03411A F279 30 01 A LEAX 1,X
03412A F27B 9F 69 A STX .BOXX1
03413A F27D 20 C8 F247 BRA CHANG4
03414 #
03415A F27F 39 E0E4 A CHANG9 RTS
03416A F280 7E E0E4 A CHANGX JMP ERR67
03417 #
03418 #
03419 ***** CHANGE COMMAND END
03421 # GET BLOCK1 COMMAND
03422 #
03423 # .COMAR+3 X1
03424 # .COMAR+5 Y1
03425 # .COMAR+7 X2
03426 # .COMAR+9 Y2

```

# FM-7 サブシステム・モニタ ソース・リスト

```

03427      *      .COMAR+11 NUMBER OF COLOR
03428      *      .COMAR+12 COLOR CODES
03429      *
03430      *
03431A F283      DOAF      A      .%ORG      SORG      LINKLN
                                SET
                                LINKLN
03432A F283      DOAF      A      PANCL      EQU      PANCL,1 NUMBER OF PICKUP COLOR
                                DOB0      A      .%ORG      .%ORG
                                .%ORG+1
03433A F283      DOB0      A      PANCL      EQU      PABCL,2 EXTENDED COLOR TABLE ADDR
                                DOB2      A      .%ORG      .%ORG
                                .%ORG+2
03434A F283      DOB2      A      PARL      EQU      PARL,1 LEFT/RIGHT FLAG (FOLLOWED BY PASCND)
                                DOB3      A      .%ORG      .%ORG
                                .%ORG+1
03435A F283      DOB3      A      PASCND      EQU      .%ORG
                                DOB4      A      .%ORG      SET
                                .%ORG+1
03436A F283      DOB4      A      PASIMG      EQU      PASIMG,1 BIT IMAGE
                                DOB5      A      .%ORG      SET
                                .%ORG+1
03437A F283      DOB5      A      PATEMP      EQU      .%ORG
                                DOB6      A      .%ORG      SET
                                .%ORG+1
03438A F283      DOB6      A      PAVB      EQU      .%ORG
                                DOB7      A      .%ORG      SET
                                .%ORG+1
03439A F283      DOB7      A      PAVR      EQU      .%ORG
                                DOB8      A      .%ORG      SET
                                .%ORG+1
03440A F283      DOB8      A      PAVG      EQU      .%ORG
                                DOB9      A      .%ORG      SET
                                .%ORG+1
03441A F283      DOB9      A      PALINK      EQU      PALINK,0 LINK ADDRESS FOR PAINT VARIABLE
                                DOB9      A      .%ORG      SET
                                .%ORG+0
03443A F283      DOB9      A      GBAOX      EQU      GBAOX,2 OFFSET LEFT POSITION
                                DOB8      A      .%ORG      SET
                                .%ORG+2

```

```

03444A F283      DOB8      A      GBDPX      EQU      GBDPX,1 OFFSET BIT NO.
                                DOB8      A      .%ORG      SET
                                .%ORG+1
03445A F283      DOB8      A      GENBX      EQU      GENBX,1 NUMBER OF BYTE
                                DOB8      A      .%ORG      SET
                                .%ORG+1
03446A F283      DOB8      A      GENX      EQU      GENX,2 NUMBER OF DOTS X
                                DOB8      A      .%ORG      SET
                                .%ORG+2
03447A F283      DOB8      A      GENY      EQU      GENY,1 NUMBER OF DOT Y
                                DOB8      A      .%ORG      SET
                                .%ORG+1
03448A F283      DOB8      A      GBI      EQU      GBI,1 TEMP
                                DOB8      A      .%ORG      SET
                                .%ORG+1
03449A F283      DOB8      A      GBJ      EQU      GBJ,1 TEMP
                                DOB8      A      .%ORG      SET
                                .%ORG+1
03450A F283      DOB8      A      GBPIX      EQU      GBPIX,2 PUT BUFFER INDEX
                                DOB8      A      .%ORG      SET
                                .%ORG+2
03451A F283      DOB8      A      GBIX      EQU      GBIX,1 CONCATENATED BIT POS.
                                DOB8      A      .%ORG      SET
                                .%ORG+1
03452A F283      DOB8      A      GBTHIX      EQU      GBTHIX,2 VRAM ADDRESS (SAVE)
                                DOB8      A      .%ORG      SET
                                .%ORG+2
03453A F283      DOB8      A      GBDIX      EQU      GBDIX,2 VRAM ADDRESS
                                DOB8      A      .%ORG      SET
                                .%ORG+2
03454A F283      DOB8      A      GBK      EQU      GBK,1 TEMP
                                DOB8      A      .%ORG      SET
                                .%ORG+1
03455A F283      DOB8      A      GBSV      EQU      GBSV,1 SHIFT VALUE
                                DOB8      A      .%ORG      SET
                                .%ORG+1
03456A F283      DOB8      A      GBCL      EQU      GBCL,3*8 EXTENDED COLORS
                                DOB8      A      .%ORG      SET
                                .%ORG+3*8

```

```

03457A F283      DOB8      A      GBDATA      EQU      GBDATA,83 AREA FOR PLASTIC
                                DOB8      A      .%ORG      SET
                                .%ORG+83
03458      *
03459A F283 7E      E0E4      A      GBLK19      JMP      ERR67
03460      *
03461      F286      A      GBLK19      EQU      #
03462A F286 8D      F2C8      A      JSR      GBLK19
03463A F286 F6      D388      A      LDB      .COMAR+11 NO. OF COLOR
03464A F286 27      F5      F283      BEQ      GBLK19 CHECK
03465A F286 C1      08      A      CMPE      #8
03466A F286 22      F1      F283      BHI      GBLK19
03467A F286 8E      D38C      A      LDX      #,COMAR+12 PTR,COLOR CODE
03468A F286 CE      D0C8      A      LDW      #GBCL COLOR TABLE
03469A F286 8D      F5      F2ED      ESR      CLEXT EXTEND COLORS
03470A F286 0F      E3      A      CLR      PASCND PICKUP CONDITION IS NORMAL.
03471      *
03472      * - OUTPUT DATA -
03473      *
03474      *
03475      *
03476A F286 BD      F3FC      A      JSR      GBLK12
03477      F29F      A      GBLK12      EQU      #
03478A F29F 96      BC      A      LDA      GENBX
03479A F2A1 97      C0      A      STA      GBI LOOP COUNTER
03480A F2A3 408E      D0E5      A      LDY      #GBDATA+2
03481A F2A7 DE      C7      A      LDW      GBDIX LOAD VRAM ADDRESS
03482      F2A9      A      GBLK13      EQU      #
03483A F2A9 8D      FC86      A      JSR      GBLKDOT
03484A F2AC E7      A0      A      STB      Y+
03485A F2AE 33      41      A      LEAU      1,U INCREMENT VRAM ADDRESS
03486A F2B0 0A      C0      A      DEC      GBI
03487A F2B2 26      F5      F2A9      ENE      GBLK13
03488      *
03489      *
03490      *
03491A F2B4 BD      F325      A      JSR      GBPLA SHIFT AND PUT
03492      *
03493A F2B7 9E      C7      A      LDX      GBDIX
03494A F2B9 30      88 50      A      LEAX      GBXS2,X INC Y CO-ORDINATE

```

```

03495A F2BC 9F      C7      A      STX      GBDIX
03496A F2BE 0A      BF      A      DEC      GENY
03497A F2C0 26      DD      F29F      A      BNE      GBLK1
03498A F2C2 0D      C4      A      TST      GBIX
03499A F2C4 27      04      F2CA      BEQ      GBLK1
03500A F2C6 96      E3      A      LDA      GBDAT
03501A F2C8 8D      3E      F308      BSR      GBDAT
03502A F2CA 39      3E      F308      GBLK17 RTS
03503      *
03504      *
03505      *
03506      *
03507      *
03508      *
03509      *
03510      *
03511A F2CB BD      E402      A      JSR      BOXIN
03512A F2CE 30      01      A      LEAX      1,X
03513A F2D0 9F      8D      A      STX      GENX
03514A F2D2 5C      0F      A      INCB
03515A F2D3 D7      BF      A      STB      GENY
03516A F2D5 CD      69      A      LOD      .BOXX
03517A F2D7 8D      E3F2      A      JSR      GYOFF
03518A F2D9 9F      B9      A      STX      GBAOX
03519A F2DB D7      BE      A      STB      GBDPX
03520A F2DE CD      60      A      LOD      .BOXX
03521A F2E0 BD      E3F2      A      JSR      GYOFF
03522A F2E2 1F      10      A      TFR      X,0
03523A F2E5 93      69      A      SUB      GBAOX
03524A F2E7 5C      0F      A      INCB
03525A F2E8 D7      BC      A      STB      GENBX
03526A F2EA 0F      C4      A      CLR      GBIX
03527A F2EC 39      3E      F308      RTS
03528      *
03529      *
03530      *
03531A F2ED D7      AF      A      CLEXT      EQU      #
03532A F2EF DF      B0      A      STB      STB
03533A F2F1 34      04      A      STU      STU
03534A F2F3 8D      0C      F301      CLEXT5      BSR      GBSC

```

```

03535A F2F5 8D      0A      F301      BSR      GBSC
03536A F2F7 80      08      F301      BSR      GBSC
03537A F2F9 30      01      A      LEAX      1,X
03538A F2FB 6A      E4      A      DEC      X
03539A F2FD 26      F4      F2F3      BNE      CLEXT5
03540A F2FF 35      84      A      RTSPUL      B
03541      *
03542      *
03543      *
03544      *
03545      *
03546      *
03547A F301 64      84      A      GBSC      EQU      #
03548A F303 56      84      A      LSR      O,X
03549A F304 1D      84      A      ROR      SEX
03550A F305 A7      C0      A      STA      ,U+
03551A F307 39      3E      F308      RTS
03552      *
03553      *
03554      *
03555      *
03556      *
03557      *
03558      *
03559      *
03560A F308 34      14      A      GBPUT      EQU      #
03561A F30A F6      D383      A      PSHS      .COMAR
03562A F30B 3E      C2      A      LDX      GBI
03563A F30D C1      7C      A      CMPE      #128-
03564A F311 25      07      F31A      BLO      GBPUT
03565A F313 5F      E0F5      A      JSR      CONTO
03566A F315 8E      0F      A      CLRB      LDY
03567A F317 8E      0F      A      CLRB      LDY
03568      *
03569A F31A A7      80      A      GBPUT1      EQU      #,COMAR
03570A F31C 5C      80      A      INCB      STA      O,X+
03571A F31D F7      D383      A      STE      .COMAR
03572A F320 9F      C2      A      STX      STX
03573A F322 35      14      A      PULS      B,X

```

```

03574A F324 39      3E      F308      RTS
03575      *
03576      *
03577      *
03578      *
03579      *
03580      *
03581      *
03582      *
03583      *
03584      *
03585A F325 96      C4      A      GBPLA      EQU      #
03586A F327 90      BB      A      GBSV=GBIX-GBOPX1
03587A F329 97      CA      A      SUBA      GBDPX
03588      *
03589A F32B 81      04      A      STA      STA
03590A F32D 2F      06      F335      ELE      GBPLA
03591A F32F 80      08      A      SUBA      #8
03592A F331 97      CA      A      STA      GBSV
03593A F333 20      08      F33D      BRA      GBPLA
03594      *
03595A F335 81      FC      A      IF GBSV-4 THEN
03596A F337 2C      04      F33D      GBPLA1      CMPA      #4
03597A F339 88      08      A      ADDA      #8
03598A F33B 97      CA      A      STA      GBSV
03599      *
03600      *
03601      *
03602A F33D 0D      CA      A      IF GBSV-4 THEN GOT
03603A F33F 28      17      F358      EMI      GBPLA
03604      *
03605A F341 96      CA      A      LDA      GBI=1 TO GBSV1
03606A F343 97      C0      A      STA      GBSV
03607A F345 27      28      F36F      GBPLA3      BEQ      GBPL10
03608      *
03609A F347 96      BC      A      LDA      LDY
03610A F349 97      C1      A      STA      GBJ
03611A F34B 8E      D0E5      A      LDY      #GBDAT
03612      *
03613      *
03614      *
03615      *
03616      *
03617      *
03618      *
03619      *
03620      *
03621      *
03622      *
03623      *
03624      *
03625      *
03626      *
03627      *
03628      *
03629      *
03630      *
03631      *
03632      *
03633      *
03634      *
03635      *
03636      *
03637      *
03638      *
03639      *
03640      *
03641      *
03642      *
03643      *
03644      *
03645      *
03646      *
03647      *
03648      *
03649      *
03650      *
03651      *
03652      *
03653      *
03654      *
03655      *
03656      *
03657      *
03658      *
03659      *
03660      *
03661      *
03662      *
03663      *
03664      *
03665      *
03666      *
03667      *
03668      *
03669      *
03670      *
03671      *
03672      *
03673      *
03674      *
03675      *
03676      *
03677      *
03678      *
03679      *
03680      *
03681      *
03682      *
03683      *
03684      *
03685      *
03686      *
03687      *
03688      *
03689      *
03690      *
03691      *
03692      *
03693      *
03694      *
03695      *
03696      *
03697      *
03698      *
03699      *
03700      *
03701      *
03702      *
03703      *
03704      *
03705      *
03706      *
03707      *
03708      *
03709      *
03710      *
03711      *
03712      *
03713      *
03714      *
03715      *
03716      *
03717      *
03718      *
03719      *
03720      *
03721      *
03722      *
03723      *
03724      *
03725      *
03726      *
03727      *
03728      *
03729      *
03730      *
03731      *
03732      *
03733      *
03734      *
03735      *
03736      *
03737      *
03738      *
03739      *
03740      *
03741      *
03742      *
03743      *
03744      *
03745      *
03746      *
03747      *
03748      *
03749      *
03750      *
03751      *
03752      *
03753      *
03754      *
03755      *
03756      *
03757      *
03758      *
03759      *
03760      *
03761      *
03762      *
03763      *
03764      *
03765      *
03766      *
03767      *
03768      *
03769      *
03770      *
03771      *
03772      *
03773      *
03774      *
03775      *
03776      *
03777      *
03778      *
03779      *
03780      *
03781      *
03782      *
03783      *
03784      *
03785      *
03786      *
03787      *
03788      *
03789      *
03790      *
03791      *
03792      *
03793      *
03794      *
03795      *
03796      *
03797      *
03798      *
03799      *
03800      *
03801      *
03802      *
03803      *
03804      *
03805      *
03806      *
03807      *
03808      *
03809      *
03810      *
03811      *
03812      *
03813      *
03814      *
03815      *
03816      *
03817      *
03818      *
03819      *
03820      *
03821      *
03822      *
03823      *
03824      *
03825      *
03826      *
03827      *
03828      *
03829      *
03830      *
03831      *
03832      *
03833      *
03834      *
03835      *
03836      *
03837      *
03838      *
03839      *
03840      *
03841      *
03842      *
03843      *
03844      *
03845      *
03846      *
03847      *
03848      *
03849      *
03850      *
03851      *
03852      *
03853      *
03854      *
03855      *
03856      *
03857      *
03858      *
03859      *
03860      *
03861      *
03862      *
03863      *
03864      *
03865      *
03866      *
03867      *
03868      *
03869      *
03870      *
03871      *
03872      *
03873      *
03874      *
03875      *
03876      *
03877      *
03878      *
03879      *
03880      *
03881      *
03882      *
03883      *
03884      *
03885      *
03886      *
03887      *
03888      *
03889      *
03890      *
03891      *
03892      *
03893      *
03894      *
03895      *
03896      *
03897      *
03898      *
03899      *
03900      *
03901      *
03902      *
03903      *
03904      *
03905      *
03906      *
03907      *
03908      *
03909      *
03910      *
03911      *
03912      *
03913      *
03914      *
03915      *
03916      *
03917      *
03918      *
03919      *
03920      *
03921      *
03922      *
03923      *
03924      *
03925      *
03926      *
03927      *
03928      *
03929      *
03930      *
03931      *
03932      *
03933      *
03934      *
03935      *
03936      *
03937      *
03938      *
03939      *
03940      *
03941      *
03942      *
03943      *
03944      *
03945      *
03946      *
03947      *
03948      *
03949      *
03950      *
03951      *
03952      *
03953      *
03954      *
03955      *
03956      *
03957      *
03958      *
03959      *
03960      *
03961      *
03962      *
03963      *
03964      *
03965      *
03966      *
03967      *
03968      *
03969      *
03970      *
03971      *
03972      *
03973      *
03974      *
03975      *
03976      *
03977      *
03978      *
03979      *
03980      *
03981      *
03982      *
03983      *
03984      *
03985      *
03986      *
03987      *
03988      *
03989      *
03990      *
03991      *
03992      *
03993      *
03994      *
03995      *
03996      *
03997      *
03998      *
03999      *

```

DEC Y LOOP	03614					* T=SUBSTR (GBDATA (GBJ+2),8,1) ;
	03615					* GBDATA (GBJ+2)=C**SUBSTR (GBDATA (GBJ+2),1,7) ;
	03616					* C=T1
REMAIN IS ZERO	03617A F34E 66	80	A	ROR	0,X	
	03618A F350 0A	C1	A	DEC	GBJ	
	03619A F352 2A	FA	F34E	BPL	GBPLA4	
	03620					END ;
	03621					END ; GOTO OUTPUT1
	03622A F354 0A	C0	A	DEC	GBI	
	03623A F356 20	ED	F345	BFA	GBPLA3	
	03624					MSHIFT ;
OF "GBLK2"	03625		F358	A	GBPLA5 EQU	
	03626					DO GBI=-1 TO GSV STEP -1
	03627A F358 96	CA	A	LDA	GESV	
	03628A F35A 97	C0	A	STA	GBI	
	03629					
	03630A F36C 9E	D0E5	A	GBPLA6 LDX	#GBDATA+2	
	03631A F36F 96	EC	A	LDA	GENEX	
	03632A F361 30	86	A	LEAX	A,X	
	03633					DO GBJ=1 TO GENEX+1
	03634					
ADDRESS,BIT NO.	03635A F363 97	C1	A	STA	GBJ	
X,ADDRESS	03636		F365	A	GBPLA7 EQU	
X,BIT NO.	03637					T=SUBSTR (GBDATA (GENEX+2-GBJ),1,1) ;
	03638					GBDATA (GENEX+2-GBJ)=SUBSTR (GBDATA (GENEX+2-GBJ),2,7)**C1
	03639					C=T1
	03640A F365 63	82	A	ROL	0,-X	
	03641					END ;
	03642A F367 0A	C1	A	DEC	GBJ	
	03643A F369 2A	FA	F365	BPL	GBPLA7	
NUMBER OF BYTE X	03644					END ;
	03645A F36B 0C	C0	A	INC	GBI	
	03646A F36D 26	ED	F35C	BNE	GBPLA6	
	03647					OUTPUT
EXTEND COLOR BIT TO A BYTE.	03648		F36F	A	GBPL10 EQU	
SET NUMBER OF COLORS FOR PICKUP PROC.	03649					P=ADDR (GBDATA (3)) ;
SET ADDRESS OF EXTEND COLORS FOR PICKUP PROC.	03650A F36F 9E	D0E5	A	LDA	#GBDATA+2	
SET LOOP COUNTER TO STACK.	03651					IF GESV+GBDPX > 7 THEN P=P+1 ;
BLUE	03652A F371 96	CA	A	LDA	GESV	
	03653A F374 96	EB	A	ADDA	GBDPX	
FED	03654A F376 81	07	A	CMPL	#7	
GREEN	03655A F378 2F	02	F37C	BLE	GBPL11	
INCREMENT ADDRESS OF ORIGINAL COLOR TABLE	03656A F37A 30	01	A	LEAX	1,X	
MORE COLOR ?	03657		F37C	A	GBPL11 EQU	
YES.	03658					IF GESV+GBDPX < 0 THEN P=P-1 ;
CLEAN UP THE STACK AND RETURN	03659A F37C 40	02	F381	TSTA		
	03660A F37D 2A	1F	A	BPL	GBPL12	
	03661A F37F 30	1F	A	LEAX	-1,X	
	03662		F381	A	GBPL12 EQU	
	03663					MASK=BAS (GBI+1)*2-1 ;
TO A BYTE	03664A F381 CE	E392	A	LDU	#BAS	
BYTE TO TABLE.	03665A F384 96	C4	A	LDA	GBIX	
	03666A F386 E6	C6	A	LDB	A,U	
PICKUP	03667A F388 58			LSLB		
EXTEND	03668A F389 5A			DECB		
SET	03669					P=-OR (P->D&MASK)* (GBDATA (1)&MASK) ;
	03670A F38A 1F	38	A	TFF	B,A	
	03671A F38C A4	84	A	ANDA	0,X	
	03672A F38E A7	84	A	STA	0,X	
	03673A F390 53			COMB		
COMM. AREA	03674A F391 D4	E3	A	ANDB	GBDATA	
DO CONTINUOUS SEND	03675A F393 E4	84	A	ORB	0,X	
BYTE.	03676A F395 E7	84	A	STB	0,X	
	03677					N=GENX+GBIX1 ;
	03678					GBIX=MOD (N,8) ;
	03679					N=N/8 ;
	03680A F397 DC	ED	A	LDD	GENX	
	03681A F399 DB	C4	A	ADDB	GBIX	
	03682A F39B 89	00	A	ADCA	#0	
	03683A F39D 1F	12	A	TFF	X,Y	
	03684A F39F 6D	E3F2	A	JSP	G10FF	ASSIGN Y TO P
	03685A F3A2 07	C4	A	STB	GBIX	
	03686A F3A4 1F	10	A	TFF	X,D	ASSIGN E TO N
	03687					DO I=1 TO N ;
	03688A F3A6 50			TSTB		
	03689A F3A7 27	08	F3B1	GBPL14 BEQ	GBPL15	
	03690					CALL
	03691A F3A9 A6	A0	A	LDA	0,Y	
	03692A F3AB ED	F308	A	JSP	GBPUT	
	03693					END ;
	03694A F3AE 5A			DECB		
	03695A F3AF 20	F6	F3A7	BFA	GBPL14	
	03696		F3B1	A	GBPL15 EQU	
	03697					GBDATA (1)=P-D ;
	03698A F3B1 A6	A4	A	LDA	0,Y	
	03699A F3B3 97	E3	A	STA	GBDATA	
	03700A F3B5 39			FTS		
	03701					
	03702					
	03703					
	03704					
	03705					GET BLOCK2 COMMAND
	03706					.COMAR+3 X1
	03707					.COMAR+5 Y1
	03708					.COMAR+7 X2
	03709					.COMAR+9 Y2
	03710					
	03711					
	03712					
	03713		F3B6	A	GBLK2 EQU	
	03714A F3B6 BD	F2CB	A	JSP	GEINT	CO-ORDINATE DATA
	03715A F3B9 3D	41	F3FC	ESF	GEINT	OUTPUT PROCESS
	03716A F3BB 9E	C7	A	LDB	GEINT	
	03717A F3BD 30	84	A	LEAX	.VB,X	
	03718A F3BF 8D	0E	F3CF	BSR	GBLI2S	
	03719A F3C1 9E	C7	A	LDB	GBDI1	
	03720A F3C3 30	39	4000	A	LEAX	.VB,X
	03721A F3C7 8D	06	F3CF	BSR	GBLI2S	
	03722A F3C9 9E	C7	A	LDB	GBDI1	
	03723A F3CB 30	39	8000	A	LEAX	.VB,X
	03724					
	03725		F3CF	A	GBLK2S EQU	
	03726A F3CF 96	BF	A	LDA	GENY	
	03727A F3D1 97	C9	A	STA	GBI	LOOP COUNTER SET
	03728		F3D3	A	GBLI2S EQU	
	03729A F3D3 9F	C5	A	STX	#GBDATA+2	
	03730A F3D5 108E	D0E5	A	LDA	GENEX	
	03731A F3D9 D6	EC	A	LDB	GENEX	
	03732A F3DB			VACON		
OF OUTPUT DATA						
COM. AREA						
PLACE OF SHIFT						
GESV=GESV-8;GOTO NEXT;END;						
GESV=GESV+8;						
MSHIFT;						

# FM-7 サブシステム・モニタ ソースリスト

```

A F3DB B5 D409 A BITA VACC VRAM READ
03733A F3DE BD E44A JSR MOVE
03734A F3E1 B7 D409 A JSR VACOFF
A F3E1 B7 D409 A JSR STA VACC
03735 F3E4 BD F325 A JSR GBPLA 1 LINE OUT
03737 F3E7 9E C5 A LDX GETMIX
03738A F3E9 30 38 50 A LEAX GBXS2,X NEXT LINE
03740A F3ED 04 C3 A DEC GBKX DEC LOOP C
03741A F3EE 26 E3 F3D3 ENE GBLK25
03742 F3F0 0D C4 A TST GB1X REMAIN ?
03744A F3F2 27 07 F3FB BEQ GBLK27
03745A F3F4 96 E3 A LDA GBDATA
03746A F3F6 BD F308 A JSR GBPUT
03747A F3F8 39 C4 A CLR GB1X REMAIN IS 0
03748A F3FB 39 C4 A GBLK27 RTS
03749 F3FC A GBLK27 RTS
03750 F3FC A GBLK27 RTS
03751A F3FC 7F D383 A CLR .COMAR+3
03752A F3FE 8E D384 A LDX .COMAR+4
03753A F402 9F C3 A STX GB1X
03754A F404 96 6C A LDX .GBY1+1 Y CO-ORDINATE
03755A F406 C6 50 A LDB #GBXS2 NUMBER OF BYTE X
03756A F408 3D 83 A MUL
03757A F409 3E B3 A LDX GBAD
03758A F40B 30 88 A LEAX D,X TOP OF DATA ADDRESS
03759A F40D 9F C7 A STX GB1X
03760A F40F C4 A CLR GB1X REMAIN, ZERO
03761A F411 39 C4 A RTS
03763 *
03764 *
03765 * POINT COMMAND
03766 *
03767 *
03768 * .COMAR+3 NUMBER OF POINT
03769 * .COMAR+4 POINT DATA
03770 *
03771 * STRUCTURE OF POINT DATA

```

```

03772 * 0 X1
03773 * 2 Y1
03774 * 4 COLOR CODE
03775 * 5 FUNCTION CODE
03776 *
03777 * TOTAL 6 BYTES
03778 *
03779 *
03780 F412 A POINT EQU *
03781A F412 B6 D383 A LDA .COMAR+3
03782A F415 27 28 F43F SEQ POIN80
03783A F417 81 14 A CMPA #20
03784A F419 22 25 F440 BHI POIN90
03785A F41B CE D384 A LDU #.COMAR+4
03786 F41E A POIN50 EQU #
03787A F41E 37 30 A PULL X,Y
03788A F420 BD E442 A JSR GAXCHK
03789A F423 37 06 A PULL D
03790A F425 C1 05 A CMPB #5
03791A F427 24 1A F443 SHS POIN92
03792A F429 DD 60 A STD .GCL
03793A F42B 34 40 A PSIB
03794A F42D 9F 64 A STX .GX
03795A F42F 109F 62 A STY .GY
03796A F432 BD E469 A JSR GPDSPI
03797A F435 BD E48F A JSR GPDSP
03798A F438 35 40 A PULS U
03799A F43A 7A D383 A DEC .COMAR+3
03800A F43D 26 DF F41E ENE POIN50
03801A F43F 39 C4 A POIN80 RTS
03802A F440 7E E0DE A POIN90 JMP ERR65
03803A F443 7E E0DE A POIN92 JMP ERR64
03804 *
03805 * POINT COMMAND END
03806 *
03807 *
03808 *
03809 *
03810 * PUT BLOCK1 COMMAND
03811 *

```

```

03812 *
03813 * .COMAR+1 PATTERN REMAINED
03814 * .COMAR+3 X1
03815 * .COMAR+5 Y1
03816 * .COMAR+7 X2
03817 * .COMAR+9 Y2
03818 * .COMAR+11 COLOR CODE
03819 * .COMAR+12 FUNCTION CODE
03820 * .COMAR+13 NUMBER OF BYTE OF PATTERN
03821 * .COMAR+14 PATTERN
03822 *
03823 *
03824 DOA0 A PENX EQU .WORK
03825 DOA2 A PENY EQU PENX+2
03826 DOA4 A PEVRAM EQU PENY+2
03827 DOA6 A PE8P EQU PEVRAM+2
03828 DOA7 A PE8P EQU PE8P+1
03829 DOA8 A PELM EQU PE8P+1
03830 DOA9 A PERM EQU PELM+1
03831 DOAA A PEDV EQU PERM+1
03832 DOAB A PE1E EQU PEDV+1
03833 DOAD A PEDH EQU PE1E+2
03834 DOAE A PERBEX EQU PEDH+1 PERB EXTEND MSB 0
03835 DOAF A PERB EQU PERBEX+1
03836 DOB0 A PENOT EQU PERB+1
03837 DOB1 A PE8AD EQU PENOT+1
03838 DOB3 A PERD EQU PE8AD+2
03839 DOB4 A PESAUE EQU PERD+1
03840 DOB5 A PE18L EQU PESAUE+1
03841 DOB7 A PE18R EQU PE18L+2
03842 DOB9 A PE18R EQU PE18R+2
03843 DOBB A PE2ENT EQU PE18R+2
03844 DOBD A PE1 EQU PE2ENT+2
03845 DOBE A PEDATA EQU PE1+1 DATA,81 BYTE
03846 *
03847 *
03848A F446 BD F446 A PELK1 EQU *
03849A F448 FC D38E A JSR .PBINT INITIALIZE
03850A F44C C1 05 A LDB .COMAR+11 A=COLOR,B=FUNCTION
03851A F44E 1022 EC89 E0DB LSH ERR64

```

```

03852A F452 8E F48E A LDX
03853A F455 ED E48E A JSR
03854 *
03855A F458 98 A3 A LDA
03856A F45A 4C A INC
03857A F45B 37 BD A STA
03858 *
03859 F45D A PB1050 EQU
03860A F460 BD F50D A LDA
03861A F462 94 BE A LDA
03862A F464 97 BE A STA
03863A F466 DE AB A LDU
03864A F468 36 AB A LDA
03865A F46A A4 C9 DOBE A ANDA
03866A F46E A7 C9 DOBE A STA
03867A F472 30 C4 A LDU
03868A F474 30 C4 A LEAX
03869A F476 90 83 A JSR
03870A F478 30 C9 4000 A LEAX
03871A F47C 90 86 A JSR
03872A F47E 30 C9 8000 A LEAX
03873A F482 30 83 A JSR
03874A F484 30 C8 50 A LEAU
03875A F486 DF A4 A STU
03876A F488 0A BD A DEC
03877A F48B 26 D0 F450 A BNE
03878A F48D 39 D0 F450 A RTS
03880 *
03881A F48E F626 A PB1FNC FDB
03882A F492 F62B A FDB
03883A F496 F62B A FDB
03884A F498 F640 A FDB
03885A F49C F641 A FDB
03886A F49E F658 A FDB
03887A F4A0 F640 A FDB
03888A F4A2 F62B A FDB

```

```

A F444 F641 A *
03887 *
03888 *
03889 * "PBINT"
03890 *
03891 * CO-ORDINATE OF
03892 * AND INITIALIZE
03893 *
03894 *
03895 *
03896 F4A6 F4A6 A PBINT EQU
03897A F4A6 BD E402 A JSR
03898A F4A9 9F A0 A STX
03899A F4AB DD A2 A STD
03900A F4AD 3E 69 A LDX
03901A F4AF 109E 66 A JSR
03902A F4B2 BD E3DC A LDB
03903A F4B5 DF A4 A STU
03904A F4B7 D6 6A A LDB
03905A F4B9 C4 07 A ANDB
03906A F4BB D7 A6 A STB
03907A F4BD CE E392 A LDU
03908A F4C0 A5 C5 A LDA
03909A F4C2 4A C5 A DECA
03910A F4C3 AB C5 A ADDA
03911A F4C5 97 A8 A STA
03912A F4C7 D6 6E A LDB
03913A F4C9 C4 07 A ANDB
03914A F4CB D7 A6 A STE
03915A F4CD AB C5 A LDA
03916A F4CF 4A C5 A DECA
03917A F4D0 43 C5 A COMA
03918A F4D1 37 A9 A STA
03919A F4D3 D6 A1 A LDB
03920A F4D5 BC A INCB
03921A F4D8 C4 07 A ANDB
03922A F4DB D7 AA A STE
03923A F4DD DC 89 A LDD
03924A F4DD DC E3F2 A JSR
03925A F4DF 34 10 A PSHS

```

```

03926A F4E1 DC 6D A LDD
03927A F4E3 BD E3F2 A JSR
03928A F4E5 1F 10 A TFR
03929A F4E8 A3 E1 A SUBD
03930A F4EA DD AB A STD
03931A F4EC 0F AD A CLR
03932A F4EE 0F AE A CLR
03933A F4F0 0F AF A CLR
03934A F4F2 0F B0 A CLR
03935A F4F4 B6 D38C A LDA
03936A F4F7 81 05 A CMPA
03937A F4F9 28 02 F4FD BNE
03938A F4FB 03 00 A COM
03939A F4FD 23 03 F502 PBINT3 BLS
03940A F4FF 7E E0DB A JMP
03941 F502 A PBINT4 EQU
03942A F502 B6 D38D A LDA
03943A F505 97 B3 A STA
03944A F507 9E D38E A LDX
03945A F50A 9F B1 A STX
03946A F50C 39 B1 A RTS
03947 *
03948 *
03949 *
03950 *
03951 * PATTERN GET
03952 *
03953 *
03954A F500 96 B4 A LDA
03955A F50F 37 BE A STA
03956A F511 DC A0 A LDD
03957A F513 93 AE A SUBD
03958A F515 2B 15 F52C BHI
03959A F517 BD E3FB A JSR
03960A F51A 8E DOBE A LDX
03961A F51B 96 AF A LDA
03962A F51F 27 02 F523 BEQ
03963A F521 30 01 A LEAX
03964A F523 3D 48 F56D PSH10 BSR
03965A F525 A7 80 A STA

```

#PB1FNC MAKE FUNCTION ENTRY  
FENTRY

PBNY+1

PB1 LOOP COUNTER SET

#  
PESHFT DATA GET & SHIFT

PBLM

PBDATA LEFT MASK

PB1B

PBRM

PBDATA,U RIGHT MASK

PBVRAM

..VB,U

ELUDSP

..VR,U

REDDSP

..VG,U

GREDSP

GXESZ,U

PBVRAM

PB1

PB1050

```

03966A F527 5A      F8 F523      DECB
03967A F528 2A      BPL
03968A F52A 37      B4 A      STA      PBSH10
03969      #          #          PSAVE  GET END,SAVE
03970      F52C      A PESH20 EQU      #
03971A F52C 36      A6 A      LDA      #BSP  SET POS.
03972A F52E 30      AD A      SUBA     PBDH  N PLACE FOR SHIFT
03973A F530 27      1F F551      BSH50
03974A F532 2B      OF F543      BSH40
03975      #          #          #
03976A F534 8E      D0BE A PESH30 LDX      #PBDATA LEFT SHIFT N PLACE
03977A F537 D6      AC A      LDB      PB1B+1
03978      F539      A PESH35 EQU      #
03979A F539 66      80 A      ROR      #,X+
03980A F53B 5A      #          #          #
03981A F53C 2A      FB F533      DECB
03982A F53E 4A      F3 F534      BPL      PESH35
03983A F53F 26      OE F551      DECA
03984A F541 20      D0BE A PESH40 BNE      PESH30
03985A F543 8E      AC A PESH41 LDX      PESH50
03986A F546 D6      AC A PESH41 LDB      PESH50
03987A F548 5C      #          #          #
03988      F549      A PESH45 EQU      #
03989A F549 63      85 A      ROL      #B,X
03990A F54B 5A      FB F549      DECB
03991A F54C 2A      FB F549      BPL      PESH45
03992A F54E 4C      F5 F546      INCA
03993A F54F 26      #          #          #
03994      F551      A PESH50 EQU      #
03995      AD A      LDA      #
03996A F551 36      AA A      ADDA     PBDH
03997A F553 36      07 A      ANDA     PBDV
03998A F555 34      AD A      ANDA     #07
04000A F557 97      07 A      STA      PBDH
04001A F55A 84      AF A      NEGA     #07
04002A F55C 37      #          #          #
04003A F55E 39      #          #          #
04004      #          #          #
04005      #          #          #

```

PB10R,PB1AND PSET

PB10R,PB1AND PRESET

PB10R,PB1NOP OR

PB1NOP,PB1AND AND

PB10R,PB1NOP EQP

PB10R,PB1AND NOT

WINDOW CHECK AND SORT  
SHIFT PROCESS

```

#
BOXINV
PBXN
PENY
..BOXX1
..BOXY1
GADDR
PBVRAM VRAM ADDRESS
..BOXX1+1
#07
PESP BIT NO,START
#BAS
B,U

```

B,U

PBLM LEFT MASK

..BOXX2+1

#07

PBER BIT NO,END

B,U

```

04006      #          #          #
04007      #          #          #
04008      #          #          #
04009      #          #          #
04010      #          #          #
04011      #          #          #
04012      #          #          #
04013      #          #          #
04014A F55F BD      E121 A PEGETO EQU      #
04015A F562 3E      D384 A JSR      CONT1 NEXT PATTERN
04016A F565 3F      B1 A      LDX      #,COMAR+4
04017A F567 B6      D383 A STX      PEEAD
04018A F56A 97      B3 A      LDA      #,COMAR+3
04019A F56C      #          #          #
04020      #          #          #
04021A F56D 34      10 A PEGETO EQU      #
04022A F56F 36      B3 A PSHS     X
04023A F571 26      07 F57A      LDA      PBDH
04024A F573 36      01 A BNE      PEGET2
04025A F575 26      E8 F55F      LDA      #,CONT
04026A F577 4F      #          #          #
04027A F578 20      08 F582      CLRA     PEGETO
04028      F57A      A PEGET2 EQU      #
04029A F57A 3E      B1 A LDX      PEEAD
04030A F57C AE      30 A LDA      #,X+
04031A F57E 3F      B1 A STX      PEEAD
04032A F580 0A      B3 A DEC      PBDH
04033A F582 0D      80 A PEGET4 TST      PENOT
04034A F584 27      01 F587      BEQ      PEGET3
04035A F586 43      #          #          #
04036A F587      #          #          #
04037      #          #          #
04038      #          #          #
04039      #          #          #
04040      #          #          #
04041      #          #          #
04042      #          #          #
04043      #          #          #

```

```

04044      #          #          #
04045      #          #          #
04046      #          #          #
04047      #          #          #
04048      #          #          #
04049      #          #          #
04050      #          #          #
04051      #          #          #
04052      #          #          #
04053      #          #          #
04054      #          #          #
04055      #          #          #
04056A F589 BD      F4A5 A PBLK2 EQU      #
04057A F58C B6      D38C A JSR      PBINT
04058A F58F 81      03 A LDA      #,COMAR+12
04059A F591 26      04 F537      CMPA     #3
04060A F593 03      A3 A BNE      PB2010
04061A F595 03      A3 A COM      PELM
04062      F597      A PB2010 EQU      #
04063A F597 8E      F66E A LDX      #PB2FNC
04064A F59A 48      36 A LSLA     A,X
04065A F59B AE      B8 A STX      PEEAD
04066A F59D 3F      A4 A LDU      PBVRAM
04067A F59F DE      0E F5B1      BSR      PB2SUB
04068A F5A1 8D      A4 A LDU      PBVRAM
04069A F5A3 DE      C9 4000 A LEAU     ..VF,U
04070A F5A5 33      06 F5B1      BSR      PB2SUB
04071A F5A9 8D      A4 A LDU      PBVRAM
04072A F5AB DE      C9 3000 A LEAU     ..VG,U
04073A F5AD 33      #          #          #
04074      #          #          #
04075      #          #          #
04076A F5B1 0F      AD A PBLK2 EQU      #
04077A F5B3 0F      AF A CLR      PBDH
04078A F5B5 36      A3 A CLR      PBER
04079A F5B7 4C      BD A LDA      PENY+1
04080A F5B9 37      #          #          #
04081      #          #          #
04082      #          #          #
04083A F5BA BD      F500 A PB2050 EQU      #

```

```

04044      #          #          #
04045      #          #          #
04046      #          #          #
04047      #          #          #
04048      #          #          #
04049      #          #          #
04050      #          #          #
04051      #          #          #
04052      #          #          #
04053      #          #          #
04054      #          #          #
04055      #          #          #
04056A F589 BD      F4A5 A PBLK2 EQU      #
04057A F58C B6      D38C A JSR      PBINT
04058A F58F 81      03 A LDA      #,COMAR+12
04059A F591 26      04 F537      CMPA     #3
04060A F593 03      A3 A BNE      PB2010
04061A F595 03      A3 A COM      PELM
04062      F597      A PB2010 EQU      #
04063A F597 8E      F66E A LDX      #PB2FNC
04064A F59A 48      36 A LSLA     A,X
04065A F59B AE      B8 A STX      PEEAD
04066A F59D 3F      A4 A LDU      PBVRAM
04067A F59F DE      0E F5B1      BSR      PB2SUB
04068A F5A1 8D      A4 A LDU      PBVRAM
04069A F5A3 DE      C9 4000 A LEAU     ..VF,U
04070A F5A5 33      06 F5B1      BSR      PB2SUB
04071A F5A9 8D      A4 A LDU      PBVRAM
04072A F5AB DE      C9 3000 A LEAU     ..VG,U
04073A F5AD 33      #          #          #
04074      #          #          #
04075      #          #          #
04076A F5B1 0F      AD A PBLK2 EQU      #
04077A F5B3 0F      AF A CLR      PBDH
04078A F5B5 36      A3 A CLR      PBER
04079A F5B7 4C      BD A LDA      PENY+1
04080A F5B9 37      #          #          #
04081      #          #          #
04082      #          #          #
04083A F5BA BD      F500 A PB2050 EQU      #

```

```

04075      #          #          #
04076A F5B1 0F      AD A PBLK2 EQU      #
04077A F5B3 0F      AF A CLR      PBDH
04078A F5B5 36      A3 A CLR      PBER
04079A F5B7 4C      BD A LDA      PENY+1
04080A F5B9 37      #          #          #
04081      #          #          #
04082      #          #          #
04083A F5BA BD      F500 A PB2050 EQU      #

```

# FM-7 サブシステム・モニタ ソース・リスト

```

04084A F56D D6 AC A LDB PB18+1
04085A F56F 9E A LDX #PB0DATA
04086A F5C2 A D409 A BITA VACC
A F5C2 B5 D409 A BITA (PB2ENT)
04087A F5C5 AD 9F D0BE A JSP VACOFF
04088A F5C9 A F5C9 B7 D409 A STA VACC
04089A F5CC 32 C8 50 A LEAU G16SZ,U
04090A F5CF 04 B0 A PBI DEC
04091A F5D1 26 E7 F56A BNE PB2050
04092A F5D3 33 A RTS
04093 A #
04094 A F504 A PB2LRM EQU #
04095A F5D4 96 A8 A LDA PELM
04096A F5D6 34 AE A ANDA PEDATA
04097A F5D8 97 BE A STA PEDATA
04098A F5DA 96 A3 A LDA PERM
04099A F5DC A4 85 A ANDA B,X
04100A F5DE A7 85 A STA B,X
04101A F5E0 39 A RTS
04102 A F5E1 A PB2SET EQU #
F1 F504 BSR PB2LRM
04103A F5E1 8D A8 A LDA PELM
04104A F5E3 96 A8 A LDA COMA
04105A F5E5 43 C4 A ANDA ,U
04106A F5E6 A4 34 A ORA ,X
04107A F5E8 AA 34 A STA ,X
04108A F5EA A7 34 A LDA PERM
04109A F5EC 36 A3 A COMA
04110A F5EE 43 C5 A ANDA B,U
04111A F5EF A4 C5 A ORA B,X
04112A F5F1 AA 85 A STA B,X
04113A F5F3 A7 85 A STA B,X
04114A F5F5 A6 85 A PB2SE5 LDA B,X
04115A F5F7 A7 C5 A STA B,U
04116A F5F9 5A DEC BPL
04117A F5FA 2A F9 F5F5 BPL PB2SE5
04118A F5FC 39 F9 F5F5 RTS
04119 A #
04120 A F5FD A PB2AND EQU #
A8 A LDA PELM
04121A F5FD 36 A8 A

```

```

04122A F5FF 9A BE A OFA PB0DATA
04123A F601 37 BE A STA PB0DATA
04124A F603 96 A3 A LDA PERM
04125A F605 AA 85 A ORA B,X
04126A F607 A7 85 A STA B,X
04127A F609 A6 85 A PB2AN5 LDA B,X
04128A F60B A4 C5 A ANDA B,U
04129A F60D A7 C5 A STA B,U
04130A F60F 5A F7 F609 BPL PB2AN5
04131A F610 2A F7 F609 BPL PB2AN5
04132A F612 39 F7 F609 RTS
04133 A #
04134 A F613 A PB2OF EQU #
F1 F504 BSR PB2LRM
04135A F613 8D B8 A PB2OF5 LDA B,X
04136A F615 A6 85 A ORA B,U
04137A F617 AA C5 A STA B,U
04138A F619 A7 C5 A STA DECB
04139A F61B 5A F7 F615 BPL PB2OF5
04140A F61C 2A F7 F615 BPL PB2OF5
04141A F61E 39 F7 F615 RTS
04142 A #
04143 A F61F A PB2EOR EQU #
B3 F504 BSR PB2LRM
04144A F61F 8D B3 A PB2EOR5 LDA B,X
04145A F621 A6 85 A ORA B,U
04146A F623 A8 C5 A STA B,U
04147A F625 A7 C5 A STA DECB
04148A F627 5A F7 F621 BPL PB2EOR5
04149A F628 2A F7 F621 BPL PB2EOR5
04150A F62A 39 F7 F621 RTS
04151 A #
04152 A #
04153 A F62B A PB1OF EQU #
AC A LDB #PB18+1
04154A F62D D6 AC A LDX
04155A F62D 108E D0BE A LDY #PB0DATA
04156A F631 A F631 B5 D409 A LNOR VACC
A F631 B5 D409 A BITA
04157A F634 A6 A5 A PB1OF5 LDA B,Y
04158A F636 AA 85 A ORA B,X
04159A F638 A7 85 A STA B,X
04160A F63A 5A DEC BPL

```

```

04161A F63B 2A F7 F634 BPL PB1OF5
04162A F63D B7 D409 A VACOFF
A F63D B7 D409 A STA VACC
04163A F640 39 A PB1NOP RTS
04164 A #
04165 A F641 A PB1AND EQU #
AC A LDB #PB18+1
04166A F641 D6 AC A LDY #PB0DATA
04167A F643 108E D0BE A LNAND VACC
04168A F647 A F647 B5 D409 A LNAND VACC
A F647 B5 D409 A BITA
04169A F64A A6 A5 A PB1AND5 LDA B,Y
04170A F64C 43 85 A ANDA B,X
04171A F64D A4 85 A STA B,X
04172A F64F A7 85 A STA DECB
04173A F651 5A F6 F64A BPL PB1AND5
04174A F652 2A F6 F64A BPL VACOFF
04175A F654 A F654 B7 D409 A STA VACC
A F654 B7 D409 A LNAND
04176A F657 39 A PB1EOR EQU #
A8 A LDB #PB18+1
04177 A #
04178 A F658 A PB1EOR5 LDA B,X
AC A LDY #PB0DATA
04179A F658 D6 AC A LNOR VACC
04180A F65A 108E D0BE A LNOR VACC
A F65A B5 D409 A LNOR VACC
04181A F65E B5 D409 A LNOR VACC
A F65E B5 D409 A BITA
04182A F661 A6 A5 A PB1EOR5 LDA B,Y
04183A F663 A8 85 A ANDA B,X
04184A F665 A7 85 A STA B,X
04185A F667 5A DEC BPL
04186A F668 2A F7 F661 BPL PB1EOR5
04187A F66A B7 D409 A VACOFF
A F66A B7 D409 A STA VACC
04188A F66D 39 A PB1FNC FDB
A F66D 39 A FDB
04189A F66E F5E1 A PB2FNC FDB
04190A F670 F5E1 A FDB
04191A F672 F5E1 A FDB
04192A F674 F5FD A FDB
04193A F676 F5F1 A FDB
04194A F678 F5E1 A FDB

```

```

04196 A #
04197 A #
04198 A #
04199 A #
04200 A #
04201 A #
04202 A #
04203 A #
04204 A #
04205 A #
04206 A #
04207 A #
04208 A #
04209 A #
04210 A #
04211 A #
04212 A #
04213 A #
04214 A #
04215 A #
04216 A #
04217 A #
04218 A #
04219 A #
04220 A #
04221 A #
04222 A #
04223 A #
04224 A #
04225 A #
04226 A #
04227 A #
04228 A #
04229 A #
04230 A #
04231 A #
04232 A #
04233 A #
04234 A #
04235A F67A 7E E0DE A #
04236 A #

```

```

04237 A #
04238 A #
04239A F67D B6 D384 A #
04240A F680 27 F8 F67A A #
04241A F682 81 0A A #
04242A F684 22 F4 F67A A #
04243 A #
04244A F686 97 AE A #
04245A F688 97 AF A #
04246 A #
04247A F68A BE D385 A #
04248A F68D 108E D387 A #
04249A F691 ED E442 A #
04250 A #
04251 A #
04252 A #
04253A F694 9F A4 A #
04254A F696 9F A0 A #
04255A F698 109F A6 A #
04256A F69E 109F AC A #
04257A F6A0 9F FF A #
04258A F6A2 97 B0 A #
04259A F6A4 97 B1 A #
04260A F6A6 86 80 A #
04261A F6A8 97 B2 A #
04262A F6AA CC 0001 A #
04263A F6AC DD A6 A #
04264 A #
04265 A #
04266 A #
04267 A #
04268A F6AD 8E D0EA A #
04269A F6B0 9F B3 A #
04270A F6B2 8E D389 A #
04271A F6B5 9F B5 A #
04272A F6B7 0F BE A #
04273 A #
04274 A #
04275 A #
04276 A #

```

```

04277A F6B9 B6 D389 A #
04278A F6BC 84 07 A #
04279A F6BE 97 A9 A #
04280A F6C0 8E D0B7 A #
04281A F6C3 04 A9 A #
04282A F6C5 56 A #
04283A F6C6 1D A #
04284A F6C7 A7 80 A #
04285A F6C9 8C D0EA A #
04286A F6CC 26 F5 F6C3 A #
04287 A #
04288 A #
04289 A #
04290 A #
04291A F6CE 8E D0C3 A #
04292A F6D1 ED F880 A #
04293A F6D4 ED F8EC A #
04294 A #
04295 A #
04296 A #
04297 A #
04298 A #
04299A F6D7 9E A4 A #
04300A F6D9 1F 10 A #
04301A F6DB C4 07 A #
04302A F6DD 27 05 F6E4 A #
04303A F6DF 07 AA A #
04304A F6E1 ED F81A A #
04305 A #
04306A F6E4 D6 AC A #
04307A F6E6 D7 AC A #
04308 A #
04309A F6E8 8E D0C3 A #
04310A F6EB ED F94D A #
04311A F6EE DC B0 A #
04312A F6F0 DD C0 A #
04313A F6F2 96 B2 A #
04314A F6F4 37 C2 A #
04315 A #
04316 A #
F6F6 A #

```



BLOCK2 END

CURSOR

.WORK

```

GCIX+2 INITIAL CO-ORDINATE Y SAVE AREA
GCII+2 X CO-ORDINATE SAVE AREA
GCX+2 Y CO-ORDINATE SAVE AREA
GCY+2 KEY CODE SAVE AREA
GCCODE+1 COLOR CODE SAVE AREA
GCCL+1 SHIFT COUNT SET AREA
GCCON+1 SHIFT DOT NUMBER
GCCONT+2 BYTE COUNT SET AREA
GCCON+1 NEEDS CO-ORDINATE NUMBER SET AREA
GCCON+1 NEEDS CO-ORDINATE NUMBER SET AREA
GCMCON+1 CURSOR SET AREA
GCCURS+3 GCAREA HEAD ADDRESS SET AREA
GCHAD+2 CENTER CO-ORDINATE SET AREA HEAD ADDRESS
GCHAD+1+2 BLUE COLOR AREA
GCBLUE+1 RED COLOR AREA
GCRE+1 GREEN COLOR AREA
GCGREE+1 BIT MARK SAVE AREA
GCBITM+1 WORK AREA
GCWOR+1 WORK AREA
GCWOR+2 RETURN FLAG
GCFLG+1 INITIAL POSITION SHIFT FLAG
GCFLG+1 INITIAL CURSOR SAVE
GCCRSI+3 SAVE AREA
GCAREA+39
ERR65

```

```

*
.COMAR+4 TEST NEEDS CO-ORDINATE NUMBER
GCERR1
#10
GCERR1
GCCON STORE NEEDS CO-ORDINATE NUMBER
GCMCON
.COMAR+5 TEST INITIAL CO-ORDINATE
.COMAR+7
GAXCHI

```

```

*
GCX SAVE X CO-ORDINATE
GCIX SAVE INITIAL CO-ORDINATE X
GCY SAVE Y CO-ORDINATE
GCII SAVE INITIAL CO-ORDINATE Y
#5FF SET CURSOR
GCCURS
GCCURS+1
#5B0
GCCURS+2
#1
GCCONT INITIAL DOT NUMBER

```

```

#GCAREA
GCHAD0
#COMAR+3
GCHAD1
GCFLG3 RETURN FLAG ON

```

CODE

.COMAR+3 LOAD COLOR CODE

```

#507
GCCL
#GCBLUE
GCCL

```

```

.Y+
#GCBLUE+3
GC0001

```

```

#GCAREA LOAD HEAD ADDRESS OF SAVE AREA
GCBDY SET BOUNDARY COUNDITION
GCSAVE VRAM SAVE

```

TO INITIAL POSITION

```

GCX LOAD X CO-ORDINATE
X,D
#507
GC00
GCCON SET SHIFT COUNT
GC5FT2
*
GCCONT+1
GCCON SET SHIFT COUNT = 1

```

```

#GCAREA LOAD HEAD ADDRESS OF GCAREA
GCDISP DISPLAY GRAPHIC CURSOR
GCCURS SAVE INITIAL CURSOR
GCCRSI
GCCURS+2
GCCRSI+2

```

```

04317A F6F6 BD FD6E A JSR KEYGET
04318A F6F9 D6 00 A LDE ..ABRT ABORT?
04319A F6F8 26 43 F740 A BNE YES GCIRQ
04320A F6F0 97 A8 A STA GCCODE SAVE KEY CODE
04321A F6FF 81 00 A CMPA #50D
04322A F701 27 15 F718 BEQ GC0004 RETURN KEY
04323A F703 81 20 A CMPA #520
04324A F705 25 4F F756 BLO GC0008
04325A F707 80 30 A SUBA #530
04326A F709 27 05 F710 BEQ GC0003
04327A F708 81 09 A CMPA #500
04328A F70D 22 E7 F6F6 BHI GC0002
04329A F70F A F70F A SKIP2
04330A F710 86 0A A GC0003 $8C CMPX #
04331A F712 97 AC A STA #10
04332A F714 97 AA A STA GCCONT+1
04333A F716 20 0E F6F6 STA GCSCON
04334 * BRA GC0002
04335 *
04336 * RETURN KEY
04337 *
04338 F718 A GC0004 EQU *
04339 *
04340 *
04341 * MEMPIZ CENTER CO-ORDINATE
04342 *
04343A F718 9E A4 A LDX GCX LOAD X CO-ORDINATE
04344A F71A 109E A6 A LDY GCY LOAD Y CO-ORDINATE
04345A F71D DE B5 A LDJ GCHAD1
04346A F71F AF C1 A STX JU++
04347A F721 10AF 01 A STY JU++
04348A F724 DF B5 A STU GCHAD1 SET CENTER CO-ORDINATE SAVE AREA
04349 *
04350 *
04351 * VRAM SAVE
04352 *
04353A F726 9E D0C3 A LDX #GCAREA LOAD HEAD ADDRESS OF SAVE DATA
04354A F729 109E B3 A LDY GCHAD0 LOAD HEAD ADDRESS OF SAVE AREA
04355 A F72C A GC0009 EQU *

```

```

04356A F72C A6 30 A LDA ,X+
04357A F72E A7 A0 A STA ,Y+ SAVE 1-BYTE DATA
04358A F730 8C D0EA A CMPX #GCAREA+39
04359A F733 26 F7 F72C A BNE GC0003
04360A F736 109F B3 A STY GCHAD0 SET END ADDRESS OF SAVE AREA
04361 *
04362A F738 0C BE A INC GCFLG3 RETURN FLAG OFF
04363A F73A 0A AE A DEC GCCON
04364A F73C 26 B8 F6F6 BNE GC0002
04365 *
04366 *
04367 * END GRAPHIC CURSOR COMMAND
04368 *
04369 F73E A GC000A EQU *
04370A F73E 20 06 F746 A GC000A EQU * GCRTS
04371 *
04372 F740 A GCIRQ EQU *
04373A F740 9E D0C3 A LDX #GCAREA
04374A F743 BD FA28 A JSR GCRSAV CURRENT CURSOR CLEAR
04375A F746 9E B3 A GCRTS LOX GCHAD0 STACKED CURSOR CLEAR
04376A F748 8C D0EA A GCIRQ5 CMPX #GCAREA
04377A F748 23 08 F755 BLS GCIRQ3 NO DATA
04378A F74D 30 88 D9 A LEAX -39,X
04379A F750 BD FA28 A JSR GCRSAV
04380A F753 20 F3 F748 A BRA GCIRQ5
04381A F755 33 * GCIRQ3 RTS
04382 * ***** END GCURS
04383 *
04384 *
04385 *
04386 *
04387 * CURSOR MOVE KEY
04388 *
04389 *
04390A F756 0D F756 A GC000B EQU *
04391A F758 26 BE A TST GCFLG3
04392A F75A 34 0A F764 A BNE GC000F
04393A F75C 8E 02 A PSHS A
04394A F75F BD FA28 A LDX #GCAREA
04395A F762 35 02 A JSR GCRSAV
04396 * PULS A

```

```

04396 F764 A GC000F EQU *
04397A F764 0F BE A CLR GCFLG3
04398A F766 81 02 A CMPA #502 GCL20
04399A F768 26 05 F76F A BNE GC0010
04400A F76A BD F83F A JSR GCL20
04401A F76D 20 1C F78B A BRA GC0019
04402A F76F 81 06 A GC0010 CMPA #508 GCR20
04403A F771 26 05 F778 A BNE GC0012
04404A F773 BD F7FC A JSR GCR20
04405A F776 20 13 F78B A BRA GC0019
04406A F778 81 0E A GC0012 CMPA #508 GCHOME
04407A F77A 26 05 F781 A BNE GC0014
04408A F77C BD F7CF A JSR GCHOME
04409A F77F 20 0A F78E A GC0019
04410A F781 80 13 A GC0014 SUBA #519
04411A F783 2B 06 F78B A BHI GC0019
04412A F785 48 *
04413A F786 8E F79A A LDX #GCTABL
04414A F789 AD 36 A JSR [A,X]
04415 *
04416A F78E 8E D0C3 A GC0013 LDX #GCAREA
04417A F78E BD F880 A JSR GCBDRY SET BOUNDARY CONDITION
04418 *
04419A F791 BD F8EC A JSR GCSAVE VRAM SAVE
04420 *
04421A F794 BD F94D A JSR GCDISP DISPLAY GRAPHIC CURSOR
04422 *
04423A F797 7E F6F6 A JMP GC0002 JUMP KEY GET
04424 *
04425 *
04426 *
04427 F79A A GCTABL EQU *
04428A F79A A FDB GCUP20 #13
04429A F79C A FDB GCDW20 #1A
04430A F79E A FDB GCNDP #1C
04431A F7A0 A FDB GCSFTR #1E
04432A F7A2 A FDB GCSFTL #1D
04433A F7A4 A FDB GCUP #1E
04434A F7A6 A FDB GCDW #1F
04435 *

```

# FM-7 サブシステム・モニタ ソース・リスト

```

04436      F7A8      A      GCU20 EQU      *
04437A     F7A8 DC   A6      A      LDD      GCY
04438A     F7AA 83   0014     A      SUBD     #20
04439A     F7AD 3A   03      F7B2     BPL      GCU29
04440A     F7AF CC   00C7     A      LDD      #GYD-1
04441A     F7B2 DD   A6      A      GCU29 STD      GCY
04442A     F7B4 39   *
04443      *
04444      *
04445      *
04446      *
04447      *
04448A     F7B5 CC   00C7     A      GCDW20 EQU      #GYD-1
04449A     F7B8 93   A6      A      SUBD     GCY
04450A     F7BA 1083 0014     A      CMPD     #20
04451A     F7BE 25   09      F7C9     BCS      GC0041
04452A     F7C0 CC   A6      A      LDD      GCY
04453A     F7C2 C3   0014     A      ADDD     #20
04454A     F7C5 DD   A6      A      STD      GCY      ADD 20-DOT TO Y CO-ORDINATE
04455A     F7C7 20   05      F7CE     BRA      GC0042
04456      F7C9     A      GC0041 EQU      #
04457A     F7C9 CC   0000     A      LDD      #00000
04458A     F7CC DD   A6      A      STD      GCY      SET Y CO-ORDINATE = 0
04459      F7CE     A      GC0042 EQU      #
04460A     F7CE 39   RTS
04461      *
04462      *
04463      *
04464      *
04465      *
04466A     F7CF 9F   A0      A      GCHOME EQU      #
04467A     F7D1 9E   A4      A      LDX      GC1X      LOAD INITIAL CO-ORDINATE X
04468A     F7D3 9E   A2      A      LDX      GC1Y      LOAD INITIAL CO-ORDINATE Y
04469A     F7D5 9F   A6      A      STX      GCY      SET INITIAL CO-ORDINATE Y
04470A     F7D7 DD   C0      A      LDD      GC0RS1
04471A     F7D9 DD   B0      A      STD      GC0URS
04472A     F7DB 36   C2      A      LDA      GC0RS1+2
04473A     F7DD 97   B2      A      STA      GC0URS+2
04474A     F7DF 39   *
04475      *

```

```

04476      *
04477      *
04478      *
04479      *
04480      *
04481      *
04482A     F7E0 DC   A6      A      GCU20 EQU      #
04483A     F7E2 3A   A6      A      SUBD     GCY      LOAD Y CO-ORDINATE
04484A     F7E4 3A   03      F7E3     BPL      GCU29
04485A     F7E6 CC   00C7     A      LDD      #GYD-1
04486      F7E9     A      GCU29 EQU      #
04487A     F7E9 DD   A6      A      STD      GCY      SET Y CO-ORDINATE = 199
04488A     F7EB 39   *
04489      *
04490      *
04491      *
04492      *
04493      *
04494      *
04495A     F7EC DC   A6      A      GCDW20 EQU      #
04496A     F7EE D3   A6      A      ADDD     GC0NT      LOAD Y CO-ORDINATE
04497A     F7F0 1083 00C7     A      CMPD     #GYD-1
04498A     F7F4 2F   03      F7F3     BLE      GCDW2
04499A     F7F6 CC   0000     A      LDD      #0
04500      F7F9     A      GCDW2 EQU      #
04501A     F7F9 DD   A6      A      STD      GCY      SET Y CO-ORDINATE = 0
04502A     F7FB 39   *
04503      *
04504      *
04505      *
04506A     F7FC CC   0014     A      GCR20 LDD      #20
04507A     F7FF D7   AA      A      STB      GCSCON
04508A     F801 D3   04      A      ADDD     GCX
04509A     F803 03   04      F809     BRA      GCRD0
04510      *
04511      *
04512      *
04513      *
04514A     F805 DC   A4      A      GCSFTR EQU      #
04515A     F807 D3   AB      A      ADDD     GC0NT

```

```

04516A     F809 DD   A4      A      GCRD0 STD      GCX
04517A     F80B 1083 027F     A      CMPD     #GYD-1
04518A     F80F 2F   09      F81A     BLE      GCSFT2
04519A     F811 CC   0000     A      LDD      #0
04520A     F814 DD   A4      A      STD      GCX      SET X CO-ORDINATE = 0
04521A     F816 C6   01      A      LDB      #1
04522A     F818 20   13      F82D     BRA      GCSFT4
04523      F81A     A      GCSFT2 EQU      #
04524A     F81A D6   AA      A      LDE      GCSCON      LOAD SHIFT COUNT
04525A     F81C 8E   00B0     A      LDX      #GCCURS
04526A     F81F A6   C2      A      GCSFT3 LDA      2,X
04527A     F821 81   FF      A      CMPA     #8FF
04528A     F823 27   08      F82D     BEQ      GCSFT4
04529A     F825 04   B0      A      LSR      GC0URS      SHIFT RIGHT 1 BIT
04530A     F827 06   C1      A      ROR      GC0URS+1
04531A     F829 06   B2      A      ROR      GC0URS+2
04532A     F82B 20   0A      F837     BRA      GCSFT5
04533      F82D     A      GCSFT4 EQU      #
04534A     F82D 86   FF      A      LDA      #8FF
04535A     F82F 97   B0      A      STA      GC0URS
04536A     F831 97   B1      A      STA      GC0URS+1
04537A     F833 86   B0      A      LDA      #80
04538A     F835 97   B2      A      STA      GC0URS+2
04539      F837     A      GCSFT5 EQU      #
04540A     F837 5A   E5      F81F     DECDB      DECB      DECREMENT SHIFT COUNT
04541A     F838 26   AC      A      LDB      GC0NT+1
04542A     F83A D8   AA      A      STB      GCSCON
04543A     F83C D7   AA      A      STB      GCSCON
04544A     F83E 39   *
04545      *
04546      *
04547A     F83F C6   14      A      GCLD0 LDB      #20
04548A     F841 D7   AA      A      STE      GCSCON
04549A     F843 DC   A4      A      LDD      GCX
04550A     F845 83   0014     A      SUBD     #20
04551A     F848 20   04      F84E     BRA      GCLD0
04552      *
04553      *
04554      *
04555      *

```

```

04556      *
04557A     F84A DC   A4      A      GCSFTL EQU      #
04558A     F84C 93   AB      A      LDD      SUEO
04559A     F84E DD   A4      A      GCLD0 STD
04560A     F850 2A   09      F85B     BPL      LDD
04561A     F852 CC   027F     A      LDD      STD
04562A     F855 DD   A4      A      STD      LDE
04563A     F857 C6   01      A      A      BEQ
04564A     F859 20   13      F86E     BRA      EQU
04565      F85B     A      F85B     A      GCSFT7 LDB
04566A     F85E D6   AA      A      LDX
04567A     F860 8E   D0B0     A      A      LDX
04568A     F862 A6   02      A      GCSFT6 LDA
04569A     F864 81   80      A      A      CMPA
04570A     F866 27   08      F86E     A      BEQ
04571A     F868 08   B2      A      A      ASL
04572A     F86A 09   B1      A      A      ROL
04573A     F86C 03   B0      A      A      ROL
04574A     F86E 20   0A      F878     BRA
04575      F86E     A      F86E     A      GCSFT8 EQU
04576A     F86E 86   01      A      A      LDA
04577A     F870 97   B0      A      A      STA
04578A     F872 86   FF      A      A      LDA
04579A     F874 97   B1      A      A      STA
04580A     F876 97   B2      F878     A      GCSFT9 EQU
04581      F878     A      F878     A      DECB
04582A     F878 5A   E5      F860     BNE
04583A     F879 26   AC      A      A      LDB
04584A     F87B D6   AA      A      A      STB
04585A     F87D D7   AA      A      A      RTS
04586      *
04587      *
04588      *
04589      *
04590      *
04591      *
04592      *
04593      *
04594      *
04595A     F880 34   10      F880     A      GCDRY EQU
04596A     F882 86   01      A      A      LDA
04597A     F884 97   AD      A      A      STA
04598A     F886 9E   A4      A      A      LDX
04599A     F888 103E A6      A      A      LDY
04600A     F88B BD   E3DC     A      A      JSR
04601      *
04602A     F88E D7   EA      A      A      STB
04603A     F890 8E   EF      A      A      STU
04604A     F892 EF   34      A      A      LDX
04605A     F894 9E   A4      A      A      LDX
04606A     F896 1F   10      A      A      TFR
04607A     F898 BD   E3FB     A      A      JSR
04608A     F89E 1F   03      A      A      TFR
04609A     F89D AE   E4      A      A      LDX
04610A     F89F 1183 0000     A      A      CMPI
04611A     F8A3 25   05      F8AE     BNE
04612A     F8A5 86   FF      A      A      LDA
04613A     F8A7 A7   02      A      A      STA
04614A     F8A9 20   04      F8AF     BRA
04615      F8AB     A      F8AB     A      GCDR1 EQU
04616A     F8AB 6F   02      A      A      CLR
04617A     F8AD CC   AD      A      A      INC
04618      F8AF     A      F8AF     A      GCDR2 EQU
04619A     F8AF 1183 004F     A      A      CMPI
04620A     F8B3 26   06      F8BE     BNE
04621A     F8B5 86   FF      A      A      LDA
04622A     F8B7 A7   03      A      A      STA
04623A     F8B9 20   04      F8BF     BRA
04624      F8BB     A      F8BB     A      GCDR3 EQU
04625A     F8BB 6F   03      F88E     CLR
04626A     F8BD CC   AD      A      A      INC
04627      F8BF     A      F8BF     A      GCDR4 EQU
04628A     F8BF 108C 0004     A      A      CMPI
04629A     F8C3 25   06      F8CB     BCS
04630A     F8C5 86   04      A      A      LDA
04631A     F8C7 A7   04      A      A      STA
04632A     F8C9 20   04      F8CF     BRA
04633      F8CB     A      F8CB     A      GCDR5 EQU
04634A     F8CB 1F   20      A      A      TFR
04635A     F8CD E7   04      A      A      STB

```

```

04636      *
04637A     F8CF A6   04      A      GCDR6 EQU
04638A     F8D1 9E   AD      A      A      LDA
04639A     F8D3 97   AD      A      A      ADDA
04640A     F8D5 CC   00C7     A      A      LDD
04641A     F8D8 93   A6      A      A      SUEO
04642A     F8DA E7   05      A      A      STB
04643A     F8DC C1   04      A      A      CMPB
04644A     F8DE 25   04      F8E4     BCS
04645A     F8E0 C6   04      A      A      LDB
04646A     F8E2 E7   05      A      A      STB
04647      F8E4     A      F8E4     A      GCDR7 EQU
04648A     F8E4 E5   05      A      A      LDB
04649A     F8E6 DE   AD      A      A      ADDB
04650A     F8E8 D7   AD      A      A      STB
04651A     F8EA 35   90      A      A      RTS
04652      *
04653      *
04654      *
04655      *
04656      *
04657      *
04658      *
04659      *
04660A     F8EC 34   10      F8EC     A      GCSAVE EQU
04661A     F8EE EE   84      A      A      PSHS
04662A     F8F0 30   06      A      A      LDU
04663A     F8F2 8D   1C      F910     A      LEAX
04664      F8F4     A      F8F4     A      BSR
04665A     F8F4 AE   E4      A      A      LDX
04666A     F8F6 EE   84      A      A      LDU
04667A     F8F8 30   88      11      A      LEAX
04668A     F8FB 33   C9      4000     A      LEAU
04669A     F8FF 8D   0F      F910     A      BSR
04670      *
04671A     F901 AE   E4      A      A      LDX
04672A     F903 EE   84      A      A      LDU
04673A     F905 30   88      1C      A      LEAX
04674A     F908 33   C9      8000     A      LEAU

```



# FM-7 サブシステム・モニタ ソース・リスト

04789A F9AC 5A	F1	F9A0	DEC6	GCD03	DECREMENT DOWN BYTE COUNT	04905		** GCRSA1	+
04790A F9AD 20	F9AF	A	BRA	+		04906			
04791	A	GCD04	LDX	#		04907	FA4C	A	GCRSA1
04792A F9AF DE	BC	A	LDU	GCDW01	LOAD VRAM ADDRESS	04908A	FA4C E5	A	EQU
04793A F9B1 3E	D0B0	A	LDX	#GCDURS	LOAD HEAD ADDRESS OF CURSOR AREA	04909A	FA4F DF	BC	VACON
04794A F9B4 80	22	A	TST	2,Y	LOAD LEFT CONDITION	04910A	FA51 10AE	62	BITA
04795A F9B6 28	06	F9BE	BNE	GCD05		04911A	FA54 A8	24	LDY
04796A F9B8 A6	84	A	LDA	0,X		04912A	FA56 C6	50	LDA
04797A F9BA A4	-1,U	A	ORA	-1,U	SET VRAM	04913A	FA58 3D		MUL
04798A F9BC A7	5F	A	STA			04914A	FA59 34	06	PSHS
04799						04915A	FA5B DC	EC	LDD
04800						04916A	FA5D A3	E1	SUBD
04801A F9BE A6	01	A	LDA	1,X	SET CENTER CURSOR	04917A	FA5F 1F	03	TFR
04802A F9C0 AA	C4	A	ORA	0,U		04918			
04803A F9C2 A7	C4	A	STA	0,U	SET VRAM	04919A	FA61 A6	24	LDA
04804						04920A	FA63 A8	25	ADDA
04805A F9C4 6D	23	A	TST	3,Y	LOAD RIGHT CONDITION	04921A	FA65 4C		INCA
04806A F9C6 26	06	F9CE	BNE	GCD06		04922		FA66	A GCRSA2
04807A F9C8 A6	02	A	LDA	2,X		04923A	FA66 E6	80	A
04808A F9CA AA	41	A	ORA	1,U		04924A	FA68 E7	C4	A
04809A F9CC A7	41	A	STA	1,U	SET VRAM	04925A	FA6A 33	C8 50	A
04810						04926A	FA6D 4A		STB
04811A F9CE	F9CE	A	GCD06	EQU		04927A	FA6E 26	F6	FA66
A F9CE B7	D409	A	VACOFF			04928			BNE
A F9CE B7	D409	A	STA	VACC		04929A	FA70 DE	6C	A
04812A F9D1 39			RTS			04930A	FA72 A6	2C	LDA
04813						04931A	FA74 81	FF	A
04814						04932A	FA76 27	04	FA7C
04815						04933A	FA78 E6	30	A
04816						04934A	FA7A E7	5F	A
04817						04935		FA7C	A GCRSA3
04818						04936A	FA7C A6	23	LDA
04819						04937A	FA7E 81	FF	A
04820						04938A	FA80 27	04	FA86
04821A F9D2	F9D2	A	GCDAND	EQU		04939A	FA82 E6	30	A
A F9D2 E5	D409	A	BITA	VACC		04940A	FA84 E7	41	A
04822A F9D5 DF	BC	A	STU	GCDW01	SAVE VRAM ADDRESS IN WORK AREA	04941		FA86	A GCRSA4
04823A F9D7 96	BA	A	LDA	GCDITH	LOAD BIT MARK	04942A	FA86 B7	D409	A
04824A F9D9 43			COMA						VACOFF
04825A F9DA 37	68	A	STA	GCDW	STORE COMPLEMENT BIT MARK				STA
04826A F9DC E6	24	A	LDB	4,Y	LOAD UPPER BYTE COUNT				
04827	F9DE	A	GCDAN1	EQU		04943A	FA89 39		RTS
04828A F9DE 5D			TSTB			04944			
04829A F9DF 27	0C	F9ED	BEQ	GCDAN2		04945			
04830A F9E1 36	BE	A	LDA	GCDW	LOAD COMPLEMENT BIT MARK	04946			
04831A F9E3 33	C8 B0	A	LEAU	-80,U	SUBTRACT 80 FROM VRAM ADDRESS	04948			
04832A F9E5 A4	C4	A	ANDA	0,U		04949			PAINT COMMAND
04833A F9E8 A7	C4	A	STA	0,U		04950			
04834A F9EA 5A			DEC6		DECREMENT UPPER BYTE COUNT	04951			
04835A F9EB 20	F1	F9DE	BRA	GCDAN1		04952			COMAR+3
04836	F9ED	A	GCDAN2			04953			COMAR+5
04837A F9ED DE	BC	A	LDU	GCDW01	LOAD VRAM ADDRESS	04954			COMAR+7
04838A F9EF E6	25	A	LDB	5,Y	LOAD DOWN BYTE COUNT	04955			COMAR+8
04839	F9F1	A	GCDAN3			04956			COMAR+9
04840A F9F1 5D			TSTB			04957A	FA8A		SOFG
04841A F9F2 27	0C	FA00	BEQ	GCDAN4		04958A	FA8A	D0B9	A
04842A F9F4 96	BE	A	LDA	GCDW	LOAD COMPLEMENT BIT MARK	04959A	FA8A	D0B9	A
04843A F9F6 33	C8 50	A	LEAU	80,U	ADD 80 AND VRAM ADDRESS	04960A	FA8A	D0B9	A
04844A F9F8 A4	C4	A	ANDA	0,U		04961A	FA8A	D0B9	A
04845A F9FE A7	C4	A	STA	0,U	SET VRAM	04962A	FA8A	D0C4	A
04846A F9FD 5A			DEC6			04963A	FA8A	D0C4	A
04847A F9FE 20	F1	F9F1	BRA	GCDAN3		04964A	FA8A	D0C4	A
04848	FA00	A	GCDAN4			04965A	FA8A	D0C4	A
04849A FA00 DE	BC	A	LDU	GCDW01	LOAD VRAM ADDRESS	04966A	FA8A	D0C4	A
04850A FA02 3E	D0B0	A	LDX	#GCDURS	LOAD HEAD ADDRESS OF CURSOR AREA	04967A	FA8A	D0C4	A
04851A FA05 E6	22	A	LDB	2,Y	LOAD LEFT CONDITION	04968A	FA8A	D0C4	A
04852A FA07 5D			TSTB			04969A	FA8A	D0C4	A
04853A FA08 26	07	FA11	BNE	GCDAN5		04970A	FA8A	D0C4	A
04854A FA0A A6	84	A	LDA	0,X		04971A	FA8A	D0C4	A
04855A FA0C 43			COMA			04972A	FA8A	D0C4	A
04856A FA0D A4	5F	A	ANDA	-1,U		04973A	FA8A	D0C4	A
04857A FA0F A7	5F	A	STA	-1,U	SET VRAM	04974A	FA8A	D0C4	A
04858	FA11	A	GCDAN5			04975A	FA8A	D0C4	A
04859A FA11 A6	01	A	LDA	1,X		04976A	FA8A	D0C4	A
04860A FA13 43			COMA			04977A	FA8A	D0C4	A
04861A FA14 A4	C4	A	ANDA	0,U		04978A	FA8A	D0C4	A
04862A FA16 A7	C4	A	STA	0,U	SET VRAM	04979A	FA8A	D0C4	A
04863						04980A	FA8A	D0C4	A
04864A FA18 E6	23	A	LDB	3,Y		04981A	FA8A	D0C4	A
04865A FA1A 5D			TSTB			04982A	FA8A	D0C4	A
04866A FA1B 26	07	FA24	BNE	GCDAN6		04983A	FA8A	D0C4	A
04867A FA1D A6	02	A	LDA	2,X		04984A	FA8A	D0C4	A
04868A FA1F 43			COMA			04985A	FA8A	D0C4	A
04869A FA20 A4	41	A	ANDA	1,U		04986A	FA8A	D0C4	A
04870A FA22 A7	41	A	STA	1,U	SET VRAM	04987A	FA8A	D0C4	A
04871	FA24	A	GCDAN6			04988A	FA8A	D0C4	A
04872A FA24			VACOFF			04989A	FA8A	D0C4	A
A FA24 B7	D409	A	STA	VACC		04990A	FA8A	D0C4	A
04873A FA27 39			RTS			04991A	FA8A	D0C4	A
04874						04992A	FA8A	D0C4	A
04875						04993A	FA8A	D0C4	A
04876						04994A	FA8A	D0C4	A
04877						04995A	FA8A	D0C4	A
04878						04996A	FA8A	D0C4	A
04879						04997A	FA8A	D0C4	A
04880						04998A	FA8A	D0C4	A
04881						04999A	FA8A	D0C4	A
04882						05000A	FA8A	D0C4	A
04883A FA28 34	10	A	GCRSAV			05001A	FA8A	D0C4	A
04884A FA2A EE	84	A	PSHS			05002A	FA8A	D0C4	A
04885A FA2C 30	06	A	LDU	0,X	LOAD VRAM ADDRESS	05003A	FA8A	D0C4	A
04886A FA2E 8D	1C	FA4C	LEAX	6,X		05004A	FA8A	D0C4	A
04887			BSR	GCRSA1		05005A	FA8A	D0C4	A
04888A FA30 AE	E4	A	LDX	0,S		05006A	FA8A	D0C4	A
04889A FA32 EE	84	A	LDU	0,X	LOAD VRAM ADDRESS	05007A	FA8A	D0C4	A
04890A FA34 30	88 11	A	LEAX	6+11,X		05008A	FA8A	D0C4	A
04891A FA37 33	C3 4000	A	LEAU	-1,VR,U	SET RED ADDRESS	05009A	FA8A	D0C4	A
04892A FA3B 8D	0F	FA4C	BSR	GCRSA1		05010A	FA8A	D0C4	A
04893						05011A	FA8A	D0C4	A
04894A FA3D AE	E4	A	LDX	0,S		05012A	FA8A	D0C4	A
04895A FA3F EE	84	A	LDU	0,X		05013A	FA8A	D0C4	A
04896A FA41 30	88 1C	A	LEAX	6+11*2,X		05014A	FA8A	D0C4	A
04897A FA44 33	C3 8000	A	LEAU	-1,VR,U	SET GREEN ADDRESS	05015A	FA8A	D0C4	A
04898A FA46 8D	02	FA4C	BSR	GCRSA1		05016A	FA8A	D0C4	A
04899A FA4A			RTSPUL			05017A	FA8A	D0C4	A
A FA4A 35	90	A	PULS	X,PC		05018A	FA8A	D0C4	A
04900						05019A	FA8A	D0C4	A
04901						05020A	FA8A	D0C4	A
04902						05021A	FA8A	D0C4	A
04903						05022A	FA8A	D0C4	A
04904						05023A	FA8A	D0C4	A

RETURN THE CONTENTS OF SAVE AREA TO VRAM

```

*
VACC
GWORK STORE VRAM ADDRESS IN WORK AREA
2,S
4,Y LOAD UPPER BYTE COUNT
#80

D
GWORK
2,Y
D,U SET TOP ADDRESS OF CURSOR
4,Y LOAD UPPER BYTE COUNT
5,Y ADD UPPER BYTE AND DOWN BYTE

*
X+
0,U
80,U ADD 80 TO VRAM ADDRESS
GCRSA2
GWORK LOAD VRAM ADDRESS
2,Y LOAD LEFT CONDITION
#5FF
GCRSA3
X+
-1,U SET VRAM
*
3,Y LOAD RIGHT CONDITION
#5FF
GCRSA4
X+
-1,U SET VRAM
*
VACC

X CO-ORDINATE
Y CO-ORDINATE
PAINT COLOR CODE
NO. OF COLOR
BOUNDARY COLORS

PALINL
PALINK
PASAVE,2 SAVE AREA
#ORG
#ORG+2
PABRAN,1 BRANCH CONDITION
#ORG
#ORG+1
PALINT,1
#ORG
#ORG+1
PAJOHO,7 CURRENT PAINT DATA
#ORG
#ORG+7
PADIRY,2 NEXT Y
#ORG
#ORG+2
PAYADD,2
#ORG
#ORG+2
PANXY,2
#ORG
#ORG+2
PACURY,2
#ORG
#ORG+2
PAOFF,1

#ORG
#ORG+1
PASTP,2 STACK PTR.
#ORG
#ORG+2
PASTAK,((PFTBL-PASTAK)/4)+4
#ORG
#ORG+((PFTBL-PASTAK)/4)+4
PADMMY,0
#ORG
#ORG+0

*
COMAR+7
GCL
INITLN
PADMMY
PASTKP
7,X A=PAINT COLOR,B=NUMBER OF BOUND COLORS.
#8 IS COLORS GE 8?
ERR69 YES, GO ERROR.
INCLUDE PAINT COLOR
8,X
8,X ADDRESS OF ORIGINAL COLORS.
#COMAR+20 EXTEND BOUNDARY COLORS BIT.
CLEXT EXTEND
INITIAL BOUNDARY POSITION.
COMAR+3
COMAR+5
GAXCHK
PAJOHO+5
PAJOHO+3
PAJOHO+4 LOAD THE Y COORDINATE.
#80 LEFT SIDE ADDR OF CURRENT Y.
PAYADD

```

```

05001A FAC4 8D FCC2 A JSR PASTP1 SEARCH RIGHT SIDE
05002A FAC7 9F BE A STX PAJOHO+1 SET RIGHT END POS.
05003A FAC9 9E C2 A LDX PAJOHO+5
05004A FACB 8D FCC7 A JSR PASTP2 SEARCH LEFT SIDE.
05005A FACE 9C BE A CMPL PAJOHO+1 IS THE INIT POS JUST ON BOUNDARY COLOR.
05006A FAD0 26 01 FAD3 BNE PA0020 YES.
05007A FAD2 39 C2 A RTS
05008A FAD3 9F C2 A PA0020 STX PAJOHO+5
05009A FAD5 86 FF A LDA #5FF
05010A FAD7 37 BD A STA PAJOHO
05011A FAD9 86 04 A LDA #4
05012A FAD8 8D FC30 A JSR PASTP1 RERUN IS RIGHT SEARCH.
05013A FADE 03 BD A COM PAJOHO SET UP-DIRECTION.
05015
05016
05017
05018
05019
0501A PAINT
0501B
0501C
0501D
0501E
0501F
05020
05021
05022
05023
05024
05025
05026
05027
05028
05029
0502A
0502B
0502C
0502D
0502E
0502F
05030
05031
05032
05033
05034
05035
05036
05037
05038
05039
0503A
0503B
0503C
0503D
0503E
0503F
05040
05041
05042
05043
05044
05045
05046
05047
05048
05049
0504A
0504B
0504C
0504D
0504E
0504F
05050
05051
05052
05053
05054
05055
05056
05057
05058
05059
0505A
0505B
0505C
0505D
0505E
0505F
05060
05061
05062
05063
05064
05065
05066
05067
05068
05069
0506A
0506B
0506C
0506D
0506E
0506F
05070
05071
05072
05073
05074
05075
05076
05077
05078
05079
0507A
0507B
0507C
0507D
0507E
0507F
05080
05081
05082
05083
05084
05085
05086
05087
05088
05089
0508A
0508B
0508C
0508D
0508E
0508F
05090
05091
05092
05093
05094
05095
05096
05097
05098
05099
0509A
0509B
0509C
0509D
0509E
0509F
050A0
050A1
050A2
050A3
050A4
050A5
050A6
050A7
050A8
050A9
050AA
050AB
050AC
050AD
050AE
050AF
050B0
050B1
050B2
050B3
050B4
050B5
050B6
050B7
050B8
050B9
050BA
050BB
050BC
050BD
050BE
050BF
050C0
050C1
050C2
050C3
050C4
050C5
050C6
050C7
050C8
050C9
050CA
050CB
050CC
050CD
050CE
050CF
050D0
050D1
050D2
050D3
050D4
050D5
050D6
050D7
050D8
050D9
050DA
050DB
050DC
050DD
050DE
050DF
050E0
050E1
050E2
050E3
050E4
050E5
050E6
050E7
050E8
050E9
050EA
050EB
050EC
050ED
050EE
050EF
050F0
050F1
050F2
050F3
050F4
050F5
050F6
050F7
050F8
050F9
050FA
050FB
050FC
050FD
050FE
050FF

05042A FB12 30 1F A LEAX -1,X NO.
05043A FB14 3F C2 A STX PAJOHO+5
05044A FB16 4F C2 A CLRA
05045A FB17 BD FC30 A JSR PASTP1 RERUN IS PAINT.
05046A FB1A 35 10 A PULS X
05047A FB1C 3F C2 A STX PAJOHO+5
05048A FB1E 9E B9 A PAPA49 LDX PASAVE
05049A FB20 3F BE A STX PAJOHO+1
05050
05051
05052
05053
05054
05055
05056
05057
05058
05059
0505A
0505B
0505C
0505D
0505E
0505F
05060
05061
05062
05063
05064
05065
05066
05067
05068
05069
0506A
0506B
0506C
0506D
0506E
0506F
05070
05071
05072
05073
05074
05075
05076
05077
05078
05079
0507A
0507B
0507C
0507D
0507E
0507F
05080
05081
05082
05083
05084
05085
05086
05087
05088
05089
0508A
0508B
0508C
0508D
0508E
0508F
05090
05091
05092
05093
05094
05095
05096
05097
05098
05099
0509A
0509B
0509C
0509D
0509E
0509F
050A0
050A1
050A2
050A3
050A4
050A5
050A6
050A7
050A8
050A9
050AA
050AB
050AC
050AD
050AE
050AF
050B0
050B1
050B2
050B3
050B4
050B5
050B6
050B7
050B8
050B9
050BA
050BB
050BC
050BD
050BE
050BF
050C0
050C1
050C2
050C3
050C4
050C5
050C6
050C7
050C8
050C9
050CA
050CB
050CC
050CD
050CE
050CF
050D0
050D1
050D2
050D3
050D4
050D5
050D6
050D7
050D8
050D9
050DA
050DB
050DC
050DD
050DE
050DF
050E0
050E1
050E2
050E3
050E4
050E5
050E6
050E7
050E8
050E9
050EA
050EB
050EC
050ED
050EE
050EF
050F0
050F1
050F2
050F3
050F4
050F5
050F6
050F7
050F8
050F9
050FA
050FB
050FC
050FD
050FE
050FF

05084A FB58 8D FB6C A JSR PALINL NO, PAINT
05085A FB5B 9F BE A STX PAJOHO+1 SET RIGHT END POS.
05086A FB5D 86 04 A LDA #4
05087A FB5F 6D FC30 A JSR RERUN IS RIGHT
05088A FB62 9E B9 A LDX PASTK1
05089A FB64 30 1F A LDX PASAVE
05090A FB66 9F C2 A LEAX -1,X
05091A FB68 03 BD A STX PAJOHO+5 SET LEFT END POS.
05092A FB6A 20 B6 FB22 A COM PAJOHO
05093A FB6C 9F 71 A PALINL STX PASEAR
05094A FB6E DC C0 A LDD PAJOHO+3
05095A FB70 DD 73 A STD .LY1
05096A FB72 DD 77 A STD .LY2
05097A FB74 8D FCC2 A JSR PASTP1
05098A FB76 34 10 A PSHS X
05099A FB78 30 1F A LEAX -1,X
0509A FB7A 9F 75 A STX .LY2
0509B FB7C 8D EF1A A JSR LINES
0509C FB7E 35 90 A RTSPL X
0509D FB80 9E B9 A PULS X,PC
0509E FB82 9E B9 A PASE40 LDX PASAVE
0509F FB84 3F BE A PASE45 STX PAJOHO+1
050A0
050A1
050A2
050A3
050A4
050A5
050A6
050A7
050A8
050A9
050AA
050AB
050AC
050AD
050AE
050AF
050B0
050B1
050B2
050B3
050B4
050B5
050B6
050B7
050B8
050B9
050BA
050BB
050BC
050BD
050BE
050BF
050C0
050C1
050C2
050C3
050C4
050C5
050C6
050C7
050C8
050C9
050CA
050CB
050CC
050CD
050CE
050CF
050D0
050D1
050D2
050D3
050D4
050D5
050D6
050D7
050D8
050D9
050DA
050DB
050DC
050DD
050DE
050DF
050E0
050E1
050E2
050E3
050E4
050E5
050E6
050E7
050E8
050E9
050EA
050EB
050EC
050ED
050EE
050EF
050F0
050F1
050F2
050F3
050F4
050F5
050F6
050F7
050F8
050F9
050FA
050FB
050FC
050FD
050FE
050FF

050A4A FB12 30 1F A LEAX -1,X NO.
050A5A FB14 3F C2 A STX PAJOHO+5
050A6A FB16 4F C2 A CLRA
050A7A FB17 BD FC30 A JSR PASTP1 RERUN IS PAINT.
050A8A FB1A 35 10 A PULS X
050A9A FB1C 3F C2 A STX PAJOHO+5
050AA FB1E 9E B9 A PAPA49 LDX PASAVE
050AB FB20 3F BE A STX PAJOHO+1
050AC
050AD
050AE
050AF
050B0
050B1
050B2
050B3
050B4
050B5
050B6
050B7
050B8
050B9
050BA
050BB
050BC
050BD
050BE
050BF
050C0
050C1
050C2
050C3
050C4
050C5
050C6
050C7
050C8
050C9
050CA
050CB
050CC
050CD
050CE
050CF
050D0
050D1
050D2
050D3
050D4
050D5
050D6
050D7
050D8
050D9
050DA
050DB
050DC
050DD
050DE
050DF
050E0
050E1
050E2
050E3
050E4
050E5
050E6
050E7
050E8
050E9
050EA
050EB
050EC
050ED
050EE
050EF
050F0
050F1
050F2
050F3
050F4
050F5
050F6
050F7
050F8
050F9
050FA
050FB
050FC
050FD
050FE
050FF

050B4A FB58 8D FB6C A JSR PALINL NO, PAINT
050B5A FB5B 9F BE A STX PAJOHO+1 SET RIGHT END POS.
050B6A FB5D 86 04 A LDA #4
050B7A FB5F 6D FC30 A JSR RERUN IS RIGHT
050B8A FB62 9E B9 A LDX PASTK1
050B9A FB64 30 1F A LDX PASAVE
050BA FB66 9F C2 A LEAX -1,X
050BB FB68 03 BD A STX PAJOHO+5 SET LEFT END POS.
050BC FB6A 20 B6 FB22 A COM PAJOHO
050BD FB6C 9F 71 A PALINL STX PASEAR
050BE FB6E DC C0 A LDD PAJOHO+3
050BF FB70 DD 73 A STD .LY1
050C0 FB72 DD 77 A STD .LY2
050C1 FB74 8D FCC2 A JSR PASTP1
050C2 FB76 34 10 A PSHS X
050C3 FB78 30 1F A LEAX -1,X
050C4 FB7A 9F 75 A STX .LY2
050C5 FB7C 8D EF1A A JSR LINES
050C6 FB7E 35 90 A RTSPL X
050C7 FB80 9E B9 A PULS X,PC
050C8 FB82 9E B9 A PASE40 LDX PASAVE
050C9 FB84 3F BE A PASE45 STX PAJOHO+1
050CA
050CB
050CC
050CD
050CE
050CF
050D0
050D1
050D2
050D3
050D4
050D5
050D6
050D7
050D8
050D9
050DA
050DB
050DC
050DD
050DE
050DF
050E0
050E1
050E2
050E3
050E4
050E5
050E6
050E7
050E8
050E9
050EA
050EB
050EC
050ED
050EE
050EF
050F0
050F1
050F2
050F3
050F4
050F5
050F6
050F7
050F8
050F9
050FA
050FB
050FC
050FD
050FE
050FF

050C4A FB58 8D FB6C A JSR PALINL NO, PAINT
050C5A FB5B 9F BE A STX PAJOHO+1 SET RIGHT END POS.
050C6A FB5D 86 04 A LDA #4
050C7A FB5F 6D FC30 A JSR RERUN IS RIGHT
050C8A FB62 9E B9 A LDX PASTK1
050C9A FB64 30 1F A LDX PASAVE
050CA FB66 9F C2 A LEAX -1,X
050CB FB68 03 BD A STX PAJOHO+5 SET LEFT END POS.
050CC FB6A 20 B6 FB22 A COM PAJOHO
050CD FB6C 9F 71 A PALINL STX PASEAR
050CE FB6E DC C0 A LDD PAJOHO+3
050CF FB70 DD 73 A STD .LY1
050D0 FB72 DD 77 A STD .LY2
050D1 FB74 8D FCC2 A JSR PASTP1
050D2 FB76 34 10 A PSHS X
050D3 FB78 30 1F A LEAX -1,X
050D4 FB7A 9F 75 A STX .LY2
050D5 FB7C 8D EF1A A JSR LINES
050D6 FB7E 35 90 A RTSPL X
050D7 FB80 9E B9 A PULS X,PC
050D8 FB82 9E B9 A PASE40 LDX PASAVE
050D9 FB84 3F BE A PASE45 STX PAJOHO+1
050DA
050DB
050DC
050DD
050DE
050DF
050E0
050E1
050E2
050E3
050E4
050E5
050E6
050E7
050E8
050E9
050EA
050EB
050EC
050ED
050EE
050EF
050F0
050F1
050F2
050F3
050F4
050F5
050F6
050F7
050F8
050F9
050FA
050FB
050FC
050FD
050FE
050FF

050D4A FB58 8D FB6C A JSR PALINL NO, PAINT
050D5A FB5B 9F BE A STX PAJOHO+1 SET RIGHT END POS.
050D6A FB5D 86 04 A LDA #4
050D7A FB5F 6D FC30 A JSR RERUN IS RIGHT
050D8A FB62 9E B9 A LDX PASTK1
050D9A FB64 30 1F A LDX PASAVE
050DA FB66 9F C2 A LEAX -1,X
050DB FB68 03 BD A STX PAJOHO+5 SET LEFT END POS.
050DC FB6A 20 B6 FB22 A COM PAJOHO
050DD FB6C 9F 71 A PALINL STX PASEAR
050DE FB6E DC C0 A LDD PAJOHO+3
050DF FB70 DD 73 A STD .LY1
050E0 FB72 DD 77 A STD .LY2
050E1 FB74 8D FCC2 A JSR PASTP1
050E2 FB76 34 10 A PSHS X
050E3 FB78 30 1F A LEAX -1,X
050E4 FB7A 9F 75 A STX .LY2
050E5 FB7C 8D EF1A A JSR LINES
050E6 FB7E 35 90 A RTSPL X
050E7 FB80 9E B9 A PULS X,PC
050E8 FB82 9E B9 A PASE40 LDX PASAVE
050E9 FB84 3F BE A PASE45 STX PAJOHO+1
050EA
050EB
050EC
050ED
050EE
050EF
050F0
050F1
050F2
050F3
050F4
050F5
050F6
050F7
050F8
050F9
050FA
050FB
050FC
050FD
050FE
050FF

050E4A FB58 8D FB6C A JSR PALINL NO, PAINT
050E5A FB5B 9F BE A STX PAJOHO+1 SET RIGHT END POS.
050E6A FB5D 86 04 A LDA #4
050E7A FB5F 6D FC30 A JSR RERUN IS RIGHT
050E8A FB62 9E B9 A LDX PASTK1
050E9A FB64 30 1F A LDX PASAVE
050EA FB66 9F C2 A LEAX -1,X
050EB FB68 03 BD A STX PAJOHO+5 SET LEFT END POS.
050EC FB6A 20 B6 FB22 A COM PAJOHO
050ED FB6C 9F 71 A PALINL STX PASEAR
050EE FB6E DC C0 A LDD PAJOHO+3
050EF FB70 DD 73 A STD .LY1
050F0 FB72 DD 77 A STD .LY2
050F1 FB74 8D FCC2 A JSR PASTP1
050F2 FB76 34 10 A PSHS X
050F3 FB78 30 1F A LEAX -1,X
050F4 FB7A 9F 75 A STX .LY2
050F5 FB7C 8D EF1A A JSR LINES
050F6 FB7E 35 90 A RTSPL X
050F7 FB80 9E B9 A PULS X,PC
050F8 FB82 9E B9 A PASE40 LDX PASAVE
050F9 FB84 3F BE A PASE45 STX PAJOHO+1
050FA
050FB
050FC
050FD
050FE
050FF

050F4A FB58 8D FB6C A JSR PALINL NO, PAINT
050F5A FB5B 9F BE A STX PAJOHO+1 SET RIGHT END POS.
050F6A FB5D 86 04 A LDA #4
050F7A FB5F 6D FC30 A JSR RERUN IS RIGHT
050F8A FB62 9E B9 A LDX PASTK1
050F9A FB64 30 1F A LDX PASAVE
050FA FB66 9F C2 A LEAX -1,X
050FB FB68 03 BD A STX PAJOHO+5 SET LEFT END POS.
050FC FB6A 20 B6 FB22 A COM PAJOHO
050FD FB6C 9F 71 A PALINL STX PASEAR
050FE FB6E DC C0 A LDD PAJOHO+3
050FF FB70 DD 73 A STD .LY1
05100 FB72 DD 77 A STD .LY2
05101 FB74 8D FCC2 A JSR PASTP1
05102 FB76 34 10 A PSHS X
05103 FB78 30 1F A LEAX -1,X
05104 FB7A 9F 75 A STX .LY2
05105 FB7C 8D EF1A A JSR LINES
05106 FB7E 35 90 A RTSPL X
05107 FB80 9E B9 A PULS X,PC
05108 FB82 9E B9 A PASE40 LDX PASAVE
05109 FB84 3F BE A PASE45 STX PAJOHO+1
0510A
0510B
0510C
0510D
0510E
0510F
05110
05111
05112
05113
05114
05115
05116
05117
05118
05119
0511A
0511B
0511C
0511D
0511E
0511F
05120
05121
05122
05123
05124
05125
05126
05127
05128
05129
0512A
0512B
0512C
0512D
0512E
0512F
05130
05131
05132
05133
05134
05135
05136
05137
05138
05139
0513A
0513B
0513C
0513D
0513E
0513F
05140
05141
05142
05143
05144
05145
05146
05147
05148
05149
0514A
0514B
0514C
0514D
0514E
0514F
05150
05151
05152
05153
05154
05155
05156
05157
05158
05159
0515A
0515B
0515C
0515D
0515E
0515F
05160
05161
05162
05163
05164
05165
05166
05167
05168
05169
0516A
0516B
0516C
0516D
0516E
0516F
05170
05171
05172
05173
05174
05175
05176
05177
05178
05179
0517A
0517B
0517C
0517D
0517E
0517F
05180
05181
05182
05183
05184
05185
05186
05187
05188
05189
0518A
0518B
0518C
0518D
0518E
0518F
05190
05191
05192
05193
05194
05195
05196
05197
05198
05199
0519A
0519B
0519C
0519D
0519E
0519F
051A0
051A1
051A2
051A3
051A4
051A5
051A6
051A7
051A8
051A9
051AA
051AB
051AC
051AD
051AE
051AF
051B0
051B1
051B2
051B3
051B4
051B5
051B6
051B7
051B8
051B9
051BA
051BB
051BC
051BD
051BE
051BF
051C0
051C1
051C2
051C3
051C4
051C5
051C6
051C7
051C8
051C9
051CA
051CB
051CC
051CD
051CE
051CF
051D0
051D1
051D2
051D3
051D4
051D5
051D6
051D7
051D8
051D9
051DA
051DB
051DC
051DD
051DE
051DF
051E0
051E1
051E2
051E3
051E4
051E5
051E6
051E7
051E8
051E9
051EA
051EB
051EC
051ED
051EE
051EF
051F0
051F1
051F2
051F3
051F4
051F5
051F6
051F7
051F8
051F9
051FA
051FB
051FC
051FD
051FE
051FF

05114A FB86 8D FC57 A PASEAL EQU #
05115A FB89 9E C2 A JSR PASEA
05116A FB8B 30 01 A LDX PAJOHO+5
05117A FB8D 9F B9 A LEAX 1,X RIGHT DOT
05118A FB8F 8D FC7 A STX PASAVE
05119A FB91 8D FC7 A JSR PASTP2 SEARCH BOUND TO LEFT
0511AA FB93 30 01 A CMPL PASAVE IS THE RIGHT DOT BOUNDARY COLOR?
0511AB FB95 26 0F FBA5 BNE PAS120 NO.
0511AC FB97 30 01 A LEAX 1,X
0511AD FB99 8D BE A LDD PAJOHO+1
0511AE FB9B 8D FCCC A JSR PASTP3 YES, SEARCH NOT BOUND TO RIGHT

```

# F-M-7 サブシステム・モニタ ソースリスト

```

05125A FB3D 9C BE A CMPX PAJOHO+1
05126A FB3E 2E 4E FBEC STXK PASTK0
05127A FBA1 2D 1E A LEAX -1,X SET RIGHT END POS
05128A FBA3 2D 3B FBEO BRA PAS145
05130A FBA5 9F B9 A PAS120 STX PASAVE
05131A FBA7 DC CA A LDD PACURY CURRENT Y
05132A FBA9 DD C6 A STD PAYADD
05133A FBAB 9E C2 A LDH PAJOHO+5
05134A FBAD 3D 1F A LEAX -1,X
05135A FBAF DC B9 A PASAVE
05136A FBB1 ED FCCD A JSR PASTP4 SEARCH NOT BOUND TO LEFT
05137A FBB4 9C B9 A CMPX PASAVE CONTINUANS?
05138A FBB6 2F 26 FBDE BLE PAS140 YES, FIND THE LEFT END POS.
05139A FBB8 9F B9 A STX PASAVE
05140A FBBB 9F 75 A STX -LX2
05141A FBBC DC C0 A LDD PAJOHO+3
05142A FBBD 73 A STD -LX1
05143A FBBD DD 77 A STD -LX2
05144A FBCE BD FCC7 A JSR PASTP2
05145A FBCE 9F C2 A STX PAJOHO+5
05146A FBCE 3D 01 A LEAX 1,X
05147A FBCE 9F 71 A STX -LX1
05148A FBCE BD EFA1 A JSR LINES PAINT
05149A FBCE BD 3E A LDA #8 RERUN IS LEFT
05150A FBCE BD FCC0 A JSR PASTK1 SAVE TO STACK
05151A FBCE 9E B9 A LDX PASAVE
05152A FBCE 3D 01 A LEAX 1,X
05153A FBCE 9F BE A STX PAJOHO+1 SET RIGHT END POINT
05154A FBCE 03 BD A COM PAJOHO
05155A FBCE 7E FB22 A JMP PASEAR
05156A FBCE 9E B9 A LDX PASAVE
05157A FBCE 9E C2 A PAS140 LDX PASAVE
05158A FBCE D6 C5 A PAS145 STX PAJOHO+5 SET LEFT END POS.
05159A FBCE D7 C1 A LDB PADIRY+1
05160A FBCE D7 C1 A STB PAJOHO+4
05161 *
05162 *
05163 *
05164 *
05165 *
05166A FBE6 9C BE A CMPX PAJOHO+1
05167A FBE8 102D FEF4 FAE0 LBLT PAPAIN
    
```

```

05168 *
05169 * TAKE OUT INFORMATION FROM STACK
05170 *
05171A FBE6 DE CD A PASTK0 EQU #
05172A FBE6 DE CD A PASTK1 EQU #
05173A FBE6 26 01 FBFB CMPLD EQU #
05174A FBE6 26 01 FBFB CMPLD EQU #
05175A FBE6 39 39 FBFB CMPLD EQU #
05176A FBE6 0F BD A PAS010 CLR PAJOHO
05177A FBE6 A6 C0 A LDA -U+
05178A FBE6 2A 02 FBFD BPL PAS020
05179A FBE6 03 BD A COM PAJOHO
05180A FBE6 39 BD A PAS020 EDRA PAJOHO
05181A FBE6 34 02 A PSHS #
05182A FBE6 84 CF A ANDA #CF
05183A FBE6 5F 5F A CLR8
05184A FBE6 44 44 A LSRA
05185A FBE6 59 59 A ROL8
05186A FBE6 44 44 A LSRA
05187A FBE6 59 59 A ROL8
05188A FBE6 C1 03 A CMPS #03 LEFT X IS -1?
05189A FBE6 25 02 FCOE BLO PAS030 NO.
05190A FBE6 C6 FF A LDB #FF YES.
05191A FBE6 D7 C2 A PAS030 STB PAJOHO+5 X2 HIGH
05192A FBE6 E6 C0 A LDB -U+ X1 LOW
05193A FBE6 DD 06 A STD PAJOHO+1 X1
05194A FBE6 A6 C0 A LDA -U+
05195A FBE6 37 C3 A STA PAJOHO+5 X2 LOW
05196A FBE6 A6 C0 A LDA -U+
05197A FBE6 97 C1 A STA PAJOHO+4 Y
05198A FBE6 DF C0 A STU PASTK1 SAVE STACK POINTER
05199A FBE6 35 02 A PULS A
05200A FBE6 84 30 A ANDA #30
05201A FBE6 44 44 A LSRA
05202A FBE6 44 44 A LSRA
05203A FBE6 44 44 A LSRA
05204A FBE6 8E 96 A LDX #PATABLE
05205A FBE6 8E 96 A JMP [A,X]
05206 *
05207A FBE6 44 44 A PATABLE EQU #
    
```

```

05208A FC2A FAE0 A FDB PAPAIN CHECK EOUNDRY COLOR
05209A FC2C FB22 A FDB PASEAR SEARH RIGHT END POINT
05210A FC2E FB66 A FDB PASEAL SEARCH LEFT END POINT
05211 *
05212 *
05213 * "PASTK1"
05214 *
05215 * SAVE INFORMATION IN STACK
05216 *
05217A FC30 A PASTK1 EQU #
05218A FC30 97 B8 A STA PABFAN
05219A FC32 DE CD A LDU PASTK1 LOAD STACK POINTER
05220A FC34 1183 D0CF A CMPLD #PASTAK
05221A FC38 27 1C FC56 BEQ PASTK1
05222 *
05223A FC3A 96 C1 A PAST10 LDA PAJOHO+4 Y
05224A FC3C A7 C2 A STA -U
05225A FC3E 96 C3 A LDA PAJOHO+6
05226A FC40 A7 C2 A STA -U
05227A FC42 96 B8 A LDA PAJOHO+2
05228A FC44 A7 C2 A STA -U
05229A FC46 96 B8 A LDA PAJOHO+1
05230A FC48 96 B8 A ORA PABFAN
05231A FC4A D6 C2 A LDB PAJOHO+5
05232A FC4C 54 A LSR8
05233A FC4D 49 A ROLA
05234A FC4E 54 A LSR8
05235A FC4F 49 A ROLA
05236A FC50 38 BD A EDRA PAJOHO
05237A FC52 A7 C2 A STA -U
05238A FC54 DF C0 A STU PASTK1
05239A FC56 39 A PASTKR RTS
05240 *
05241 *
05242 * "PASEA"
05243 *
05244 * SEARCH CONTINUOUS END POINT
05245 *
05246A FC57 A PASEA EQU #
05247A FC57 DE C0 A LDU PAJOHO+3
    
```

```

05248A FC59 33 41 A
05249A FC5B 06 C1 A
05250A FC5D 96 BD A
05251A FC5F 26 04 FC65
05252A FC61 5A 5E A
05253A FC62 33 5E A
05254A FC64 A FC64
05255A FC65 5C 21 A PASE10
05256A FC66 C1 C8 A
05257A FC68 25 05 FC6F
05258A FC6A 32 62 A
05259A FC6C 7E FBEC A
05260A FC6E 86 C8 A PASE20
05261A FC71 3D 50 A
05262A FC72 DD C5 A
05263A FC74 DD C6 A
05264A FC76 DF C4 A
05265A FC78 86 50 A
05266A FC7A D6 C1 A
05267A FC7C 3D CA A
05268A FC7D DD CA A
05269A FC7F 39
05270 *
05271 *
05272 *
05273 *
05274 *
05275 *
05276 *
05277 *
05278 *
05279 *
05280 *
05281 *
05282 *
05283 *
05284 *
05285 *
05286 *
05287 *
    
```

```

05288 *
05289A FC80 DE C6 A PABGET
05290A FC82 96 CC A
05291A FC84 33 CC A
05292A FC86 B5 D409 A GBLKDT
05293A FC88 12 A FC86 B5
05294A FC8A E6 C9 8000 A
05295A FC8E 07 B8 A
05296A FC90 E6 C9 4000 A
05297A FC94 A6 C4 A
05298A FC96
05299A FC98 B7 D409 A
05300A FC9A DD B5 A
05301A FC9D 36 AF A
05302A FC9F 9E B0 A
05303A FCA1 D6 B6 A PABG50
05304A FCA3 E8 80 A
05305A FCA5 D7 B5 A
05306A FCA7 D6 B7 A
05307A FCA9 E8 80 A
05308A FCAB DA B5 A
05309A FCAD D7 B5 A
05310A FCAF D6 B8 A
05311A FCB1 E8 80 A
05312A FCB3 D6 B5 A
05313A FCB5 53 B4 A
05314A FCB6 DA B4 A
05315A FCB8 D7 B4 A
05316A FCBA 4A B4 A
05317A FCBB 26 E4 FCA1
05318A FCBD D8 B3 A
05319A FCBF D7 B4 A
05320A FCC1 39
05321A FCC2 CE 0100 A PASTP1
05322A FCC5 20 12 FC03
05323A FCC7 CE FF00 A PASTP2
05324A FCCA 20 1E FCEA
05325A FCCC CE 01FF A PASTP3
    
```

```

05326A FCCC CE 01FF A PASTP3
05327A FCCF C3 0008 A
05328A FCD2 B5 E3FB A
05329A FCD5 C1 50 A
05330A FCD7 2F 13 FCEC
05331A FCD9 C6 50 A PASTRM
05332A FCDB 20 0F FCEC
05333A FCDD CE FFFF A PASTP4
05334A FCE0 83 0008 A
05335A FCE3 BD E3FB A
05336A FCE5 C1 FF A
05337A FCE8 2C 02 FCEC
05338A FCEA C6 FF A PASTLM
05339A FCEC DF B2 A PA.SEA
05340A FCEE D7 BC A
05341A FCF0 8C FFFF A
05342A FCF3 3F FD34
05343A FCF5 8C 0280 A
05344A FCF8 2C 31 FD2B
05345A FCFA 1F 10 A
05346A FCFC C4 07 A
05347A FCFE 34 04 A
05348A FDF0 1F 82 A
05349A FDF2 DD E3FB A
05350A FDF5 D7 CC A
05351A FDF7 BD FC80 A
05352A FDF8 35 04 A
05353A FDFC 8E E392 A
05354A FDFE E6 B2 A
05355A FD01 1F 82 A
05356A FD03 2B 03 FD18
05357A FD05 58
05358A FD06 5A
05359A FD07
05360A FD08
05361A FD09 5A 8C A PA.S20
05362A FD10 53
05363A FD1A D4 B4 A
05364A FD1C D7 B4 A
05365A FD1E 26 FD3F
    
```

```

LEAU 1,U
LDB PAJOHO+4
LDA PAJOHO
BNE PASE10
DECB
LEAU -2,U
SKIP1
FCB $21
INCB
CMPE #6VD
ELO PASE20
LEAS 2,S
JMP PASTKD
LDA #80
MUL
STD PANXTY
STD PAYADD
STU PADIRY
LDB PAJOHO+4
MUL
STD PACURY
RTS

```

```

# 'PABGET'
# INPUT AND REFERENCE
# PAYADD, PAXOFF, PANBCL
# PABCLE, PASCND
# 'GELI.DT' IS USED BY GETBLOCK1.
# U REG IS ADDRESS OF VRAM, AND THAT IS
# NOT DESTROYED BY THIS PROGRAM.
# OUTPUT
# PAGING SEARCHING COLORS IMG.
# ACCE IS SAME IT.
# WORK
# PATEMP, PAVE, PAVF, PAVG

```

```

EQU #
LDU PAYADD
LDA PAXOFF
LEAU A,U
VACON
BITA VACC
NOP
LDB ..V6,U
STB PAVG
LDB ..VR,U
LDA ..V6,U
VACOFF
STA VACC
STD PAVE
CLR PAGING
LDA PANBCL
LDX PABCLE
LDB PAVE
EORB X+
STB PATEMP
LDB PAVR
EORB X+
ORB PATEMP
STB PATEMP
LDB PAVG
EORB X+
ORB PATEMP
COMB
ORB PAGING
STB PAGING
DECA
BNE PABG50
EORB PASCND
STB PAGING
RTS
LDU #0100 TO RIGHT,BOUND
ERA PASTRM
LDU #0F00 TO LEFT,BOUND
ERA PASTLM
LDU #01FF TO RIGHT,NOT BOUND

```

```

ADDD #8
JSR LSRD3
CMPE #80
BLE PA,SEA
LDB #30
BRA PA,SEA
LDU #0FFFF TO LEFT, NOT BOUND
SUEO #8
JSR LSRD3
CMPE #1
EGE PA,SEA
LDB #1
STU PARL
STB PALINT
CMPX #1
BLE PA,LMX
CMPX #GXD
EGE PA,RMX
X,D
ANDE #007
PSHS B
TFR X,D
JSR LSRD3
STB PAXOFF
JSR PABGET
PULS B
LDX #EAS
LDB B,X
LDA PARL
BMT PA,S20
LSLB
DECB
SKIP2
FCB #3C CMPX #
DECB
COMB
ANDE PASTMG
STB PASTMG
ENE PA,S80

```

```

05366A FD20 96 CC A PA,S25 LDA PAXOFF
05367A FD22 06 B2 A LDB PAPL
05368A FD24 28 09 FD2F BMT PA,S30
05369A FD26 4C EC A INCA
05370A FD27 91 0D FD38 CMPA
05371A FD29 2D 0D FD38 BLT
05372A FD2B 8E 0280 A PA,FMX LDX #GXD
05373A FD2E 39 CC A RTS
05374A FD2F 4A PA,S30 DECA
05375A FD30 91 EC A CMPA
05376A FD32 2E 04 FD38 PALINT
05377A FD34 9E 04 FD38 BLT
05378A FD37 39 FFFF A PA,LMX LDX #1
05379A FD38 97 CC A PA,S40 STX
05380A FD3A 8D FC30 A JSR
05381A FD3D 27 E1 FD20 PAXOFF
05382A FD3F 36 08 A PA,S80 SEQ
05383A FD41 06 CC A LDB
05384A FD43 3D 05 A MUL
05385A FD44 1F 01 A TFR
05386A FD46 06 B4 A LDB
05387A FD48 96 B2 A LDA
05388A FD4A 28 07 FD53 BMT
05389A FD4C 58 CC A PA,S83 LSLB
05390A FD4D 25 0B FD5A ECS
05391A FD4F 30 01 A ERASE
05392A FD51 20 F3 FD4C LEAX
05393A FD53 30 1F A PA,S85 LEAX
05394A FD55 54 05 A LSRB
05395A FD56 24 F8 FD53 BCC
05396A FD58 30 08 A BCC
05397A FD5A 23 PA,S30 RTS
05398A FD5B 25 PA,RTS RTS

```

```

05400 # INKEY COMMAND
05401 # INKEY EQU
05402 # INKEY EQU
05403 # INKEY EQU
05404 # INKEY EQU
05405 # INKEY EQU
05406 # INKEY EQU

```

```

05407 # THE BIT1 DIRECT TO ERASE KEY BUFFER.
05408 # THE BIT2 IS SAME AS THE BIT1.
05409 # WHEN WAIT IS REQUESTED, KEY IS UNLOCKED.
05410 #
05411A FD0C 06 D383 A LDA CHDPRM
05412A FD0E 35 02 A BITA CHDPRM
05413A FD0F 27 02 FD05 SEQ INKEY
05414A FD10 8D 33 FD3E BSR INKEY
05415A FD12 74 D383 A INKEY LSR
05416A FD14 8D 06 FD70 BSR INKEY
05417A FD16 FD D383 A STD CHDPRM
05418A FD18 39 FTS

```

```

05419 05420A FD6E 1A 01 A IEYGET SEC
05421 05422A FD70 24 04 FD76 IEYIN EQU
05423A FD72 36 00 A BCC
05424A FD74 97 02 A LDA
STA
# INPUT
# C FLAG : WAIT FOR KEY DATA IF C IS SET.
# OUTPUT
# ACCA : KEY CODE
# ACCE : 0 MEANS NO DATA

```

```

05433A FD76 1A 40 A IEYIN0 SEF
05434A FD78 08 04 A ORCC
05435A FD7A 26 04 A LDB
05436A FD7C 24 14 FD86 BNE
05437A FD7E D6 00 A BCC
05438A FD80 26 10 FD92 LDB
05439A FD82 3C BF A BNE
05440A FD84 2A F0 FD76 CLF
ANDCC
BRA
KEYIN0
WAIT.
05442A FD86 0A 04 A KEYIN1 DEC
05443A FD88 9E 05 A LDX
I,HEAD
DECREMENT COUNTER OF KEY BUFFER.
LOAD THE CURRENT BUFFER HEAD.

```

```

05444A FD8A A6 80 A LDA
05445A FD8C 8D 07 FD95 X,+
05446A FD8E 9F 05 A KEYIN2 STX
05447A FD90 C6 01 A LDB
05448A FD92 3C BF A KEYIN3 CLF
05449A FD94 39 EF A ANDCC
RTS
05451A FD95 8C D380 A INCBX
05452A FD97 26 03 FD9D CMPX
05453A FD99 8E D360 A BNE
05454A FD9B 39 D360 A INCBX
05455A FD9D 3C BF A KEYIN3
05456A FD9F 1A 40 A SEF
05457A FDA0 0F 04 A ORCC
05458A FDA2 8E D360 A CLR
05459A FDA4 9F 05 A LDX
05460A FDA6 9F 07 A STX
05461A FDA8 1A EF A STX
05462A FDAE 39 EF A CLF
ANDCC
RTS
05464 05465A FDC4 34 FD4C A KEYIFQ EQU
05466A FDC6 FC D400 A PSYS
05467A FDC8 2C 04 FD8D LDD
05468A FDCB 36 05 FD8D BMT
05469A FDCD 26 02 FD89 A LDB
05470A FDE0 8D E5 FD3E BNE
05471A FDE2 8D 2B FDE6 BSR
05472A FDE4 20 26 FDE3 BSR
05473A FDE6 34 04 A KEYIR1 PSYS
05474A FDE8 6D FE3D A JSR
05475A FDEA 25 04 A PULS
05476A FDEC A6 80 A LDA
05477A FDEE 19 FDE1 A BMT
05478A FDE8 27 19 FDE3 BCC
05479A FDEA 34 02 A PSYS
05480A FDEC 96 03 A LDB
05481A FDEE 26 06 FDD8 BNE

```



# FM-7 サブシステム・モニタ ソース リスト

```

054834 FDD0 34 10 A PSMS X SAVE STRING PTR
054844 FDD2 80 CA FDBE ESP IBEFEA YES, ERASE KEY BUFFER BEFORE SET CMC
054854 FDD4 35 10 A PULS X RELOAD STRING PTR
054864 FDD6 E6 30 A KEYIR3 LDB ,X+
054874 FDD8 0C 0C FDE6 ESP ISTEUF
054884 FDDA 6A E4 DEC ,S
054894 FDDC 26 F3 FDD6 KEYIR3
054904 FDE0 35 0C A PULS A CLEAN UP THE STACK.
054914 FDE0 A SI IP2
054924 FDE1 07 09 A KEYIR3 FCB #C CMPX #
054934 FDE3 35 56 A KEYIR3 STE #PFIRQ SET PF NO.
054944 FDE5 36 A KEYIR3 PULS D,X,U. RELOAD REGISTERS.
054954 FDEB 36 02 A KSTBU3 LDA HLOC1 IS KEY LOCKED?
054964 FDEB 36 17 FE01 ENE ISTEUB3 YES.
054974 FDEB 36 04 A LDA KCOUNT IS THE BUFFER FULL?
054984 FDEB 36 20 A CMPA #EMAX
054994 FDEE 24 11 FE01 EHS ISTEUB3 YES.
055004 FDF0 0C 04 A INC KCOUNT INCREMENT COUNTER.
055014 FDF2 06 07 A LDA LDB LOAD THE CURRENT STORE ADDRESS.
055024 FDF4 E7 00 A STB ,U+ STORE DATA TO BUFFER.
055034 FDF6 1183 D380 A CMPU #BFEND INCREMENT PTR.
055044 FDF8 26 03 FDF6 ENE ISTEUB3 (KEY BUFFER IS CIRCULAR.)
055054 FDFC CE D360 A LDU #EVBUFF
055064 FDFD DF 07 A KSTBUS STU I.TAIL
055074 FDFD DF FE01 A KSTBUS EQU I
055084 FE01 35 RTS
05511 0020 A KEMAX EQU KBFEND-KEYBUF
05512 I FUNCTION KEY COMMAND
05514 I
05515 FE02 A FUNLEY EQU I
05516A FE02 40 A SEF #MASI
05517A FE04 F8 D383 A ORCC #400
05518A FE07 27 28 FE31 ENE FUNE9 IS THE KEY NUMBER .GT. ZERO?
05519A FE09 C1 0A A CMPB #10
05520A FE0B 22 24 FE31 ENE FUNE9 YES, IS THE KEY NUMBER .LE. 10?
05521A FE0D 66 D384 A LDA #COMAR+4
05522A FE10 81 0F A CMPA #15
05523A FE12 22 1D FE31 ENE FUNE9

```

```

05503A FE87 44 02 FE8C
05504A FE88 24 02 FE8C
05505A FE8A E7 A4 A
05506A FE8C 31 21 A SETTM1
05507A FE8E C6 04 A
05508A FE90 44 A
05509A FE91 24 08 FE9B
05510A FE93 EE 84 A
05511A FE95 EF A4 A
05512A FE97 EE 02 A
05513A FE99 EF 22 A
05514A FE9B 30 04 A SETTM5
05515A FE9D 31 24 A
05516A FE9F 5A A
05517A FE9F EE FE90
05518A FE9F 20 0E FE92
05519 I
05520 I FEAD
05521 I
05522A FE9F 03 FE44 A REDTIM
05523A FE9F 03 0A A
05524A FE9F 0E D00B A
05525A FE9F 103E D384 A
05526A FE9F 11 A
05527A FE9F 6D E44A A
05528 I
05529 FE82 A TMCT
05530A FE82 03 0A A
05531A FE84 27 08 FE8E
05532A FE86 A
05533A FE88 1A 40 A
05534A FE88 8D 0E FE8C
05535A FE8A A
05536A FE8A 1C BF A
05537A FE8A 0F 0A A
05538A FE8E 39 A
05539 I TMCT99
05540 I
05541 I 20MS
05542 I
05543 I
05544 FE8F A TIMM1

```

```

05524A FE14 8D 27 FE3D ESR FUNADF
05525A FE16 1F 89 A TFP A,E
05526A FE18 58 LSL LSL
05527A FE1A 63 84 A LSL
05528A FE1C 56 80 A POPB
05529A FE1E E7 80 A STB ,X+
05530A FE20 05 0F A LDB #PFS2-1
05531A FE22 80 12 FE34 BSR CLPAPA
05532A FE24 40 52 FE77 BEQ PFINDC
05533A FE26 CE D385 A LDU #COMAR+5
05534A FE28 CE FE28 A FUNE5
05535A FE2A E6 C0 A STB ,U+
05536A FE2C E7 80 A STB ,X+
05537A FE2E 4A 0A DECA
05538A FE30 26 F9 FE28 ENE FUNE5
05539A FE32 20 46 FE77 ENE PFINDC
05540A FE34 7E E0E7 A FUNE9 JNP ERF68
05541 I
05542 FE34 A CLPAPA EQU I
05543 FE34 10 A PSMS X
05544A FE36 6F 80 A CLPAP5 CLP
05545A FE38 5A DEC B
05546A FE3A 26 FB FE36 ENE CLPAP5
05547A FE3C 35 90 A PTSPUL X
05548A FE3E 35 90 A PULS X,PC
05549 I
05550A FE3D 58 LSL LSL
05551A FE3F 58 LSL LSL
05552A FE41 58 LSL LSL
05553A FE43 58 LSL LSL
05554A FE45 3A D2E0 A LDH #,PFTBL-PFS2
05555A FE47 3A ABLX
05556A FE49 39 RTS
05557 I
05558 I GET STRING OF FUNCTION-KEY COMMAND
05559 I
05560 FE48 A GSTRPF EQU I
05561 FE48 D383 A LDU #,COMAR+3 I LOAD COMMON ADDRESS
05562A FE4E CE D383 A LDB ,U+
05563A FE4E E6 C0 A

```

```

05541A FE8F 36 0A A
05542A FE91 26 02 FE8C
05543A FE93 8D 03 FE8C
05544A FE95 0F 0A A TIMRTI
05545A FE97 38 A
05546 I
05547 I TIMER
05548 I
05549 FE83 A TINEP
05550A FE85 96 1C A
05551A FE87 27 1B FE87
05552A FE89 28 19 FE87
05553A FE8B 0A 1D A
05554A FE8D 26 15 FE87
05555A FE8E A6 3F D05A A
05556A FE8F E6 3F D05C A
05557A FE91 03 1E A
05558A FE93 27 02 FE80
05559A FE95 C8 08 A
05560A FE97 E0 E518 A CURTM5
05561A FE99 38 14 A
05562A FE9B 37 1D A
05563A FE9D FE87 A CURTM5
05564A FE9F 108E 0000 A
05565A FE8B 3E D010 A
05566A FE8E CE FF68 A
05567A FE91 C0 04 A
05568A FE93 6C 32 A TM020
05569A FE95 A6 84 A
05570A FE97 A1 C0 A
05571A FE99 25 0E FF05
05572A FE9B 6F 84 A
05573A FE9D 5A F3 FE83
05574A FE9F 26 08 A
05575A FF00 36 08 A
05576A FF02 85 08 A
05577A FF04 27 03 FF05
05578A FF06 31 A8 40 A
05579 I
05580A FF09 36 08 A TM100

```

```

05564A FE4E 27 E4 FE31 BEQ GSTRP3
05565A FE4D C1 0A A CMPE #10
05566A FE4F 02 50 FE31 GSTRP3
05567A FE51 8D EA FE3D ESR FUNADF
05568A FE53 E6 80 A LDB ,X+
05569A FE55 C4 7F A ANDB #57F
05570A FE57 86 10 A LDA #PFS2
05571A FE59 E7 C0 A GSTRP5
05572A FE5B E9 80 A STB ,U+
05573A FE5D 4A 0A DECA
05574A FE5E 26 F9 FE59 ENE GSTRP5 I DECREMENT
05575A FE5F 39 RTS
05576 FE5F FE31 A GSTRP5 EQU FUNE9
05577 I
05578 I INTERRUPT CONTROL FOR PROGRAMABLE FUNCTION KEY
05579 I
05580 FE61 A PFINT EQU I
05581A FE61 40 A SEF #40
05582A FE63 FC D383 A ORCC #400
05583A FE65 8E D2C0 A LDD CMOPRM
05584A FE67 8E 84 A PFINTS LDX #,PFTBL
05585A FE69 44 84 A PFINTS LSL
05586A FE6B 56 84 A POPB
05587A FE6D 66 84 A POPB
05588A FE6F 36 88 10 A LEAX #,PFTBL
05589A FE71 8C D380 A CMPE #,PFTBL
05590A FE73 25 F2 FE69 BLO PFINT5
05591A FE75 1C BF A PFINDC
05592A FE77 7E E0E7 A ANDCC #SEF
05593 I INDICT DISPLAY PF STRINGS.
05594 I
05595 I SET TIMER COMMAND
05596 I
05597 I
05598 FE7C A SETTIM EQU I
05599A FE7C 0A A COM #,TMAC NMI MASK
05600A FE7E 8E D383 A LDX #CMOPRM
05601A FE81 108E D00B A LDU #,STC
05602A FE85 EC 81 A LDD ,X++

```

```

05581A FF08 85 01 A
05582A FF0D 27 0F FF1E
05583A FF0F 3E 0C A
05584A FF11 9C 10 A
05585A FF13 26 03 FF1E
05586A FF15 3E 0E A
05587A FF17 9C 12 A
05588A FF19 26 03 FF1E
05589A FF1B 31 A8 10 A
05590 I
05591A FF1E 0C 16 A TM200
05592A FF20 C3 FFFF A
05593A FF23 0D 16 A
05594A FF25 25 08 FF2F
05595A FF27 3E 14 A
05596A FF29 30 1F A
05597A FF2B 3F 14 A
05598A FF2D 20 1F FF4E
05599A FF2F 26 10 FF4E TM220
05600A FF31 0C 14 A
05601A FF33 26 19 FF4E
05602A FF35 0C 13 A
05603A FF37 0D 14 A
05604A FF39 0C 1A A
05605A FF3B 0D 16 A
05606A FF3D 36 16 A
05607A FF3F 85 02 A
05608A FF41 27 08 FF4E
05609A FF43 31 A8 20 A
05610A FF45 85 04 A
05611A FF47 27 04 FF4E
05612A FF49 88 02 A
05613A FF4B 97 06 A
05614 I
05615A FF4E 86 D381 A
05616A FF51 D6 09 A
05617A FF53 27 04 FF59
05618A FF55 84 F0 A

```

```

LSRA
BCC SETTM1
STB .Y
LEAY 1,Y
LDB #4
LSRA
BCC SETTM5
LDU .X
STU .Y
LDU 2,X
STU 2,Y
LEAY 4,X
LEAY 4,Y
DECB
BNE SETTM3
BRA TMCT

```

## TIMER COMMAND

```

EQU *
COM .TMAC
LDX #.STC
LDY #CMDPRM+1
LDB #1+4+4
JSR MOVE

EQU *
COM .TMAC
BEQ TMCT99
SEF
ORCC #540
BSR TIMER NMI
CLF
ANDCC #5BF
CLR .TMAC
RTS

```

## TIMER

```

EQU *

```

## PACKAGE

```

LDA .TMAC
BNE TIMRT1
BSR TIMER
CLR .TMAC
RTI

EQU *
LDX .CURSR
BEQ CURTM3
BNI CURTM3
OCC .CURTM
BNE CURTM3
LDA (.CBAD)
LDB (.CAAD)
COM .CURST
BEQ CURTM3
FDB .AHNV
JSR CDSPC
LDA #20
STA .CURTM
EQU *
LDY #0
LDX #.ST1+4
LDU #TMDAT
LDB #4
INC .-X
LDA .X
CMPA .U+
BLO TM100
CLR .X
DECB
BNE TM020
LDA .STC
BITA #NEWDAT
BEQ TM100
LEAY NWD1,Y

LDA .STC

```

```

BITA #TMTCT
BEQ TM200
LDX .ST1
CMPX .ST11
BNE TM200
LDX .ST1+2
CMPX .ST11+2
BNE TM200
LEAY TM1,Y

LDD .ST2+2
ADD0 #5FFF
STD .ST2+2
BCS TM220
LDX .ST2
LEAX -1,X
STX .ST2
BRA TM300
BNE TM300
LDD .ST2
BNE TM900
LDD .ST20
STD .ST2
LDD .ST20+2
STD .ST2+2
LDA .STC
BITA #TMTCT
BEQ TM900
LEAY IN1,Y
BITA #TMTCT
BEQ TM900
EORA #TMTCT
STA .STC

EQU *
LDA STATUS
LDB .PFIRQ
BEQ TM310
ANDA #PF15

```

```

05720A FF57 0F 09 A CLR .PFIRQ
05721A FF59 34 20 A TM910 PSHS Y
05722A FF5B EA 81 A ORB ORB 1,S OR TIMER FLAGS.
05723A FF60 27 0A FF69 BEQ TM990 NO SYSREQ EXIST.
05724A FF6F 34 02 A PSHS A
05725A FF61 EA E0 A ORB .S+
05726A FF63 F7 D381 A STB STATUS SET INTERRUPT FLAGS
05727A FF66 65 D404 A SYSREQ # SYSTEM REQUEST
A FF66 B5
05728A FF69 35 A0 A TM990 RTSPUL Y
A FF69 35
05729A FF66 32 A TMDAT FCB 50,60,60,24
A FF6C 3C A
A FF6D 3C A
A FF6E 18 A
05731 0001 A TMTCT EQU $01
05732 0002 A TMTCT EQU $02
05733 0004 A TMTCT EQU $04
05735 0008 A NEWDAT EQU $08
05737 *
05738 * TEST COMMAND
05739 *
05740 * THIS COMMAND IS ABLE TO REFER TO THE
05741 * SUB-SYSTEM'S AREA IS ADDRESSED BY ABSOLUTE
05742 * ADDRESSING.
05743 *
05744 * TEST SUB COMMAND
05745 *
05746 * $90 RETURN TO SYSTEM
05747 * $91 MOVE DATA
05748 * $92 BRANCH TO SUB COMMAND
05749 * $93 BRANCH TO MACHINE LANGUAGE SUBROUTINE
05750 *
05751 * OTHER CODE GIVE "INVALID COMMAND" ERROR.
05752 *
05753 A TEST EQU *
05754A FF6F 8E FF6F D38E EQU LDX #CMDPRM+8
05755 0001 A IFX 1

```

```

05756 ENDC
05757A FF72 A6 80 A TSTCMD LDA .X+ YES, WHAT IS SUBCMD.?
05758A FF74 80 90 A SUBA #530
05759A FF76 2B 0B FF69 EMI TSTERR
05760A FF78 27 0C FF66 BEQ TSTRTS RETURN
05761A FF7A 4A 18 FF65 DECA
05762A FF7B 27 18 FF65 BEQ TSTMOV MOVE
05763A FF7D 4A 18 FF65 DECA
05764A FF7E 27 18 FF65 BEQ TSTBRA BRANCH
05765A FF80 4A 18 FF65 DECA
05766A FF81 27 08 FF66 BEQ TSTBSR CALL MACHINE
05767A FF83 7E E0ED A TSTERR JMP ERR70 INVALID COMMAND.
05768A FF86 39 E0ED A TSTRTS RTS RETURN TO SYSTEM
05769 *
05770A FF87 AE 84 A TSTERA LDX .X BRANCH ADDRESS LOAD.
05771A FF89 20 E7 FF72 BRA TSTCMD
05772 *
05773A FF8B EE 31 A TSTBSR LDU .X++ MACHINE ADDRESS LOADED.
05774A FF8D 34 10 A PSHS X SAVE SUBCMD. PTR.
05775A FF8F AD C4 A JSR .U SUBROUTINE CALL.
05776A FF91 35 10 A PULS X RESTORE SUBCMD. PTR..
05777A FF93 20 D0 FF72 BRA TSTCMD
05778A FF95 10AE 31 A TSTMOV LDY .X++ LOAD SOURCE ADDRESS.
05779A FF98 EE 31 A LDU .X++ LOAD DESTINATION ADDR..
05780A FF9A EC 31 A LDD .X++ LOAD BYTE COUNT.
05781A FF9C 27 04 FF72 BEQ TSTCMD NO MOVED DATA.
05782A FF9E 34 10 A PSHS X SAVE
05783A FFA0 1F 01 A TFR D,X
05784A FFA2 A6 A0 A TSTMPLP LDA .Y+
05785A FFA4 A7 C0 A STA .U+
05786A FFA6 30 1F A LEAX -1,X FINISH?
05787A FFA8 26 FFA2 BNE TSTMPLP NO, GO DO MORE.
05788A FFAA 35 10 A PULS X RESTORE
05789A FFAC 20 C4 FF72 BRA TSTCMD
05790 0001 A IFX 1
05791 ENDC
05793A FFF0 *
05794 ORG $FFFO
05795A FFF0 E000 A FDB RESTAR RESERVED.
05796A FFF2 E000 A FDB RESTAR SWI3

```

```

05797A FFF4 E000 A FDB RESTAR SWI2
05798A FFF6 FDAC A FDB KEYIRQ FIRO
05799A FFF8 E06E A FDB IRQAB IRQ ABEND
05800A FFFA E000 A FDB RESTAR SWI
05801A FFFC FEEF A FDB TIMNMI NMI 20MS TIMER
05802A FFFE E000 A FDB RESTAR RESET
05803 *
05804 END

```

# FM-7 BIOSソース・リスト

00001		NAM	FBIOS		00115		# B0 DISK SIDE
00002		OPT	CAT,LL=120,REL		00116		#
00003					00117		# B7 - 1
00004P 0000		PSCT	FBASIC PROGRAM SECTION		00118		#
00005					00119		#
00006		XREF	ABTCHK		00120		# :: RCBUNT
00007	0313	A ABORTF	EQU \$313		00121		#
00008		XDEF	CBIOS		00122		# B1 - 0
00009					00123		# B7 - 2
00010	0000	A FBASIC SET	0 ASSEMBLE FBASIC BIOS		00124		#
00011	00FF	A FM_LCM SET	255		00125		#
00012					00126		# RCBMTB
00013		TTL	BIOS FOR FBASIC		00127		#
00014					00128		*****
00015					00130		# VECTOR ADDRESS
00016					00131		#
00017					00132	FFFE	A PRESET EQU
00018			FUJITSU PERSONAL COMPUTER		00133	FFFC	A PNMI EQU
00019			BASIC INPUT/OUTPUT SYSTEM		00134	FFFA	A PSWI EQU
00020			FOR FBASIC		00135	FFF8	A PIRQ EQU
00021			DESIGNED BY R.K.		00136	FFF6	A PFIRQ EQU
00022			CODED BY I.I.		00137	FFF4	A PSW12 EQU
00023			DATE AUGUST '82		00138	FFF2	A PSW13 EQU
00024					00139	FFF0	A PVCT01 EQU
00025					00140	FFEE	A SINMAX EQU
00026					00141	FFED	A DSTDRV EQU
00028					00142	FFEB	A DHOTDR EQU
00029	0000	A BOOT SET	0 NO=0 BOOTSTRAP LOADER.		00143	FFEA	A CURTR8 EQU
00030					00144	FFED	A CRORIV EQU
00031					00145		#
00032					00146		# SHARED MEMORY E
00033					00147		#
00034					00148	FC80	A SHARED EQU
00035		BIOS EQU	BIOS ENTRY POINT		0000		A SMSTA EQU
00036					0001		A SMCFG EQU
00037					0002		A SMCMD EQU
00038		JSP	0,1 EACH DRIVEP		0003		A SMDAT EQU
00039					00152		
00040			PROCESSING		00153	0080	A SMEND EQU
00041					00154		#
00042					00155	000C	A FF EQU
00043		RTS	RETURN		000A		A LF EQU
00044					00156		A ESC EQU
00045					00157	001B	A INKEY EQU
00046					00158	0029	A PUT EQU
00047					00159	0003	A GET EQU
00048					00160	0004	A GETC EQU
00049					00161	0005	A GETB EQU
00050					00162	001B	A GETB1 EQU
00051			1. SET PCB CONTENTS		00163		#
00052			2. LDH #PCB ADDRESS		00164		# :: I/O AREA MA
00053			3. JSP (BIOS)		00165		#
00054			4.C (CARRY) CHECK		00166		# \$FD00
00055					00167		# . (READ)
00056					00168		# B7
00057					00169		# B6-1
00058					00170		# B0
00059					00171		#
00060					00172		# WRITE
00061					00173		# B7
00062					00174		# B6
00063	0000	A RQNO EQU	0 REQUEST CODE		00175		# B5
00064	0001	A RCBSTA EQU	1 STATUS		00176		# B4-2
00065	0002	A RCBDBA EQU	2 BUFFER ADDRESS		00177		# B1
00066	0002	A RCBCLN EQU	2 ANALOG PORT CHANNEL NUMBER.		00178		# B0
00067	0002	A MOTOFF EQU	2 CASSETTE MOTOR ON/OFF FLAG.		00179		#
00068	0002	A CRWDAT EQU	2 CASSETTE READ/WRITE DATA.		00180		# \$FD01
00069					00181		# B7
00070	0003	A PCBFRG EQU	3 ANALOG PORT VOLTAGE RANGE.		00182		# B6
00071					00183		# B5
00072	0004	A PCBCTD EQU	4 HARD COPY COLOR DATA.		00184		# B4
00073	0004	A RCBLNH EQU	4 NUMBER OF BYTES (H)		00185		# B3
00074	0004	A PCBTRI EQU	4 TRACK NUMBER		00186		# B2
00075	0004	A RCBPAH EQU	4 PAGE ADDRESS (H)		00187		# B1
00076	0004	A RCBAPD EQU	4 ANALOG PORT INPUT DATA.		00188		# B0
00077	0004	A RCBJCD EQU	4 KANJI JIS CODE.		00189		#
00078					00190		# \$FD02
00079	0005	A RCBCLN EQU	5 NUMBER OF BYTES (L)		00191		# (READ)
00080	0005	A RCBCT EQU	5 SECTOR NUMBER		00192		# B7
00081	0005	A RCBPAL EQU	5 PAGE ADDRESS (L)		00193		# B6
00082					00194		# B5
00083	0006	A RCBSD EQU	6 DISK SIDE		00195		# B4
00084	0006	A RCBPCH EQU	6 PAGE COUNTS (H)		00196		# B3
00085	0006	A RCBPMH EQU	6 MAX BUFFER LENGTH (H)		00197		# B2
00086					00198		# B1
00087	0007	A RCBUNT EQU	7 DRIVE UNIT		00199		# B0
00088	0007	A RCBPCL EQU	7 PAGE COUNTS (L)		00200		#
00089	0007	A RCBML EQU	7 MAX BUFFER LENGTH (L)		00201		# WRITE
00090	0007	A RCBMTB EQU	7 SUBIN MAX TRANSFER BYTES.		00202		# B7
00091					00203		# B6
00092					00204		# B5
00093					00205		# B4-3
00094			RCECNL : ANALOG PORT CHANNEL NO 0-3.		00206		# B2
00095			MOTOFF : FF=ON,00=OFF		00207		# B1
00096					00208		# B0
00097			PCBFRG : "H" HIGH RANGE "L" LOW RANGE.		00209		#
00098					00210		#
00099					00211		# \$FD03
00100			B7 - 0 TRACK NUMBER \$00 - \$4C		00212		# B7
00101					00213		# B6
00102					00214		# B5-1
00103					00215		# B0
00104					00216		#
00105			B7 - 0 SECTOR NUMBER \$01 - \$1A		00217		*****
00106					00218		#
00107					00219		# \$FD04
00108					00220		# B7
00109					00221		# B6-3
00110			B7 (MSE) BUBBLE CASSETTE NUMBER		00222		# B2
00111			B6 - 0 PAGE ADDRESS (H)		00223		# B1
00112					00224		# B0
00113					00225		#
00114					00226		# \$FD05
					00227		# B7

0 SIDE 0  
1 SIDE 1  
DON'T CARE

DRIVE UNIT NO.  
DON'T CARE

SUBIN TRANSFER BYTES 0 - 128.

\*\*\*\*\*

EQU DEFINITION

\$FFFE RESET VECTOR  
\$FFFC NMI VECTOR  
\$FFFA SWI VECTOR  
\$FFF8 IRQ VECTOR  
\$FFF6 FIQ VECTOR  
\$FFF4 SWI2 VECTOR  
\$FFF2 SWI3 VECTOR  
\$FFF0 NOT IN USE  
\$FFE8 SUB WORK MAX BUFFER CHECK.  
\$FFE6  
\$FFE4 MOTOR FLAG AREA.  
\$FFE2 CURRENT TRACK ADDRESS.  
\$FFE0 CURRENT DRIVE.

DEFINITION

\$FC80  
0  
1  
2  
3

128

\$0C FORM FEED  
\$0A LINEFEED  
\$18 ESCAPE  
\$29 INKEY  
\$03 PUT  
\$04 GET  
\$05 GETC  
27 GET BLOCK 1

CASSETTE,LP OUTPUT/KEY DATA,CLOCK INDICATOR

DATA 8(KEY DATA)  
NOT IN USE  
CLOCK INDICATOR

SLCTIN (LP)  
STFB (LP)  
RESERVED  
NON CONNECTION  
AUDIO CASSETTE REMOTE CONTROL  
MIC (CASSETTE)

LP DATA OUTPUT,KEY DATA INPUT  
DATA 7  
DATA 6  
DATA 5  
DATA 4  
DATA 3  
DATA 2  
DATA 1  
DATA 0

CASSETTE,LP INPUT/IRQ ENABLE REGISTER

FEAD DATA (CASSETTE)  
NOT IN USE  
DET2 (LP)  
DET1 (LP)  
PE (LP)  
ACKING (LP)  
ERROR (LP)  
BUSY (LP)

SYNDET (PS232C)  
RXPDY (PS232C)  
TXPDY (PS232C)  
NOT IN USE  
TIMER  
LP  
KEY

BELL OUTPUT  
BELL ON/OFF  
BELL (SHORT SOUND)  
NON CONNECTION  
SPEAKER SWITCH

\*\*\*\*\*

SUBSYSTEM INPUT  
BUSY  
NOT IN USE  
PARITY ERROR  
STOP KEY INDICATOR  
ATTENTION

SUBSYSTEM OUTPUT  
HALT / SUB BUSY

00028  
00029  
00030  
00031  
00032  
00033  
00034  
00035  
00036  
00037  
00038  
00039  
00040  
00041  
00042  
00043  
00044  
00045  
00046  
00047  
00048  
00049  
00050  
00051  
00052  
00053  
00054  
00055  
00056  
00057  
00058  
00059  
00060  
00061  
00062  
00063  
00064

00065  
00066  
00067  
00068  
00069  
00070  
00071  
00072  
00073  
00074  
00075  
00076  
00077  
00078  
00079  
00080  
00081  
00082  
00083  
00084  
00085  
00086  
00087  
00088  
00089  
00090  
00091  
00092  
00093  
00094  
00095  
00096  
00097  
00098  
00099  
00100  
00101

00102  
00103  
00104  
00105  
00106  
00107  
00108  
00109  
00110  
00111  
00112  
00113  
00114  
00115  
00116  
00117  
00118  
00119  
00120  
00121  
00122  
00123  
00124  
00125  
00126  
00127  
00128  
00129  
00130  
00131  
00132  
00133  
00134  
00135  
00136  
00137  
00138  
00139  
00140  
00141  
00142  
00143  
00144  
00145  
00146  
00147  
00148  
00149  
00150  
00151  
00152  
00153  
00154  
00155  
00156  
00157  
00158  
00159  
00160  
00161  
00162  
00163  
00164  
00165  
00166  
00167  
00168  
00169  
00170  
00171  
00172  
00173  
00174  
00175  
00176  
00177  
00178  
00179  
00180  
00181  
00182  
00183  
00184  
00185  
00186  
00187  
00188  
00189  
00190  
00191  
00192  
00193  
00194  
00195  
00196  
00197  
00198  
00199  
00200  
00201  
00202  
00203  
00204  
00205  
00206  
00207  
00208  
00209  
00210  
00211  
00212  
00213  
00214  
00215  
00216  
00217  
00218  
00219  
00220  
00221  
00222  
00223  
00224  
00225  
00226  
00227  
00228  
00229  
00230  
00231  
00232  
00233  
00234  
00235  
00236  
00237  
00238  
00239  
00240  
00241  
00242  
00243  
00244  
00245  
00246  
00247  
00248  
00249  
00250  
00251  
00252  
00253  
00254  
00255  
00256  
00257  
00258  
00259  
00260  
00261  
00262  
00263  
00264  
00265  
00266  
00267  
00268  
00269  
00270  
00271  
00272  
00273  
00274  
00275  
00276  
00277  
00278  
00279  
00280  
00281  
00282  
00283  
00284  
00285  
00286  
00287  
00288  
00289  
00290  
00291  
00292  
00293  
00294  
00295  
00296  
00297  
00298  
00299  
00300  
00301  
00302  
00303  
00304  
00305  
00306  
00307  
00308  
00309  
00310  
00311  
00312  
00313  
00314  
00315  
00316  
00317  
00318  
00319  
00320  
00321  
00322  
00323  
00324  
00325  
00326  
00327  
00328  
00329  
00330  
00331  
00332  
00333  
00334  
00335  
00336  
00337  
00338  
00339  
00340  
00341  
00342  
00343  
00344  
00345  
00346  
00347  
00348  
00349  
00350  
00351  
00352  
00353  
00354  
00355  
00356  
00357  
00358  
00359  
00360  
00361  
00362  
00363  
00364  
00365  
00366  
00367  
00368  
00369  
00370  
00371  
00372  
00373  
00374  
00375  
00376  
00377  
00378  
00379  
00380  
00381  
00382  
00383  
00384  
00385  
00386  
00387  
00388  
00389  
00390  
00391  
00392  
00393  
00394  
00395  
00396  
00397  
00398  
00399  
00400  
00401  
00402  
00403  
00404  
00405  
00406  
00407  
00408  
00409  
00410  
00411  
00412  
00413  
00414  
00415  
00416  
00417  
00418  
00419  
00420  
00421  
00422  
00423  
00424  
00425  
00426  
00427  
00428  
00429  
00430  
00431  
00432  
00433  
00434  
00435  
00436  
00437  
00438  
00439  
00440  
00441  
00442  
00443  
00444  
00445  
00446  
00447  
00448  
00449  
00450  
00451  
00452  
00453  
00454  
00455  
00456  
00457  
00458  
00459  
00460  
00461  
00462  
00463  
00464  
00465  
00466  
00467  
00468  
00469  
00470  
00471  
00472  
00473  
00474  
00475  
00476  
00477  
00478  
00479  
00480  
00481  
00482  
00483  
00484  
00485  
00486  
00487  
00488  
00489  
00490  
00491  
00492  
00493  
00494  
00495  
00496  
00497  
00498  
00499  
00500  
00501  
00502  
00503  
00504  
00505  
00506  
00507  
00508  
00509  
00510  
00511  
00512  
00513  
00514  
00515  
00516  
00517  
00518  
00519  
00520  
00521  
00522  
00523  
00524  
00525  
00526  
00527  
00528  
00529  
00530  
00531  
00532  
00533  
00534  
00535  
00536  
00537  
00538  
00539  
00540  
00541  
00542  
00543  
00544  
00545  
00546  
00547  
00548  
00549  
00550  
00551  
00552  
00553  
00554  
00555  
00556  
00557  
00558  
00559  
00560  
00561  
00562  
00563  
00564  
00565  
00566  
00567  
00568  
00569  
00570  
00571  
00572  
00573  
00574  
00575  
00576  
00577  
00578  
00579  
00580  
00581  
00582  
00583  
00584  
00585  
00586  
00587  
00588  
00589  
00590  
00591  
00592  
00593  
00594  
00595  
00596  
00597  
00598  
00599  
00600  
00601  
00602  
00603  
00604  
00605  
00606  
00607  
00608  
00609  
00610  
00611  
00612  
00613  
00614  
00615  
00616  
00617  
00618  
00619  
00620  
00621  
00622  
00623  
00624  
00625  
00626  
00627  
00628  
00629  
00630  
00631  
00632  
00633  
00634  
00635  
00636  
00637  
00638  
00639  
00640  
00641  
00642  
00643  
00644  
00645  
00646  
00647  
00648  
00649  
00650  
00651  
00652  
00653  
00654  
00655  
00656  
00657  
00658  
00659  
00660  
00661  
00662  
00663  
00664  
00665  
00666  
00667  
00668  
00669  
00670  
00671  
00672  
00673  
00674  
00675  
00676  
00677  
00678  
00679  
00680  
00681  
00682  
00683  
00684  
00685  
00686  
00687  
00688  
00689  
00690  
00691  
00692  
00693  
00694  
00695  
00696  
00697  
00698  
00699  
00700  
00701  
00702  
00703  
00704  
00705  
00706  
00707  
00708  
00709  
00710  
00711  
00712  
00713  
00714  
00715  
00716  
00717  
00718  
00719  
00720  
00721  
00722  
00723  
00724  
00725  
00726  
00727  
00728  
00729  
00730  
00731  
00732  
00733  
00734  
00735  
00736  
00737  
00738  
00739  
00740  
00741  
00742  
00743  
00744  
00745  
00746  
00747  
00748  
00749  
00750  
00751  
00752  
00753  
00754  
00755  
00756  
00757  
00758  
00759  
00760  
00761  
00762  
00763  
00764  
00765  
00766  
00767  
00768  
00769  
00770  
00771  
00772  
00773  
00774  
00775  
00776  
00777  
00778  
00779  
00780  
00781  
00782  
00783  
00784  
00785  
00786  
00787  
00788  
00789  
00790  
00791  
00792  
00793  
00794  
00795  
00796  
00797  
00798  
00799  
00800  
00801  
00802  
00803  
00804  
00805  
00806  
00807  
00808  
00809  
00810  
00811  
00812  
00813  
00814  
00815  
00816  
00817  
00818  
00819  
00820  
00821  
00822  
00823  
00824  
00825  
00826  
00827  
00828  
00829  
00830  
00831  
00832  
00833  
00834  
00835  
00836  
00837  
00838  
00839  
00840  
00841  
00842  
00843  
00844  
00845  
00846  
00847  
00848  
00849  
00850  
00851  
00852  
00853  
00854  
00855  
00856  
00857  
00858  
00859  
00860  
00861  
00862  
00863  
00864  
00865  
00866  
00867  
00868  
00869  
00870  
00871  
00872  
00873  
00874  
00875  
00876  
00877  
00878  
00879  
00880  
00881  
00882  
00883  
00884  
00885  
00886  
00887  
00888  
00889  
00890  
00891  
00892  
00893  
00894  
00895  
00896  
00897  
00898  
00899  
00900  
00901  
00902  
00903  
00904  
00905  
00906  
00907  
00908  
00909  
00910  
00911  
00912  
00913  
00914  
00915  
00916  
00917  
00918  
00919  
00920  
00921  
00922  
00923  
00924  
00925  
00926  
00927  
00928  
00929  
00930  
00931  
00932  
00933  
00934  
00935  
00936  
00937  
00938  
00939  
00940  
00941  
00942  
00943  
00944  
00945  
00946  
00947  
00948  
00949  
00950  
00951  
00952  
00953  
00954  
00955  
00956  
00957  
00958  
00959  
00960  
00961  
00962  
00963  
00964  
00965  
00966  
00967  
00968  
00969  
00970  
00971  
00972  
00973  
00974  
00975  
00976  
00977  
00978  
00979  
00980  
00981  
00982  
00983  
00984  
00985  
00986  
00987  
00988  
00989  
00990  
00991  
00992  
00993  
00994  
00995  
00996  
00997  
00998  
00999  
01000  
01001  
01002  
01003  
01004  
01005  
01006  
01007  
01008  
01009  
01010  
01011  
01012  
01013  
01014  
01015  
01016  
01017  
01018  
01019  
01020  
01021  
01022  
01023  
01024  
01025  
01026  
01027  
01028  
01029  
01030  
01031  
01032  
01033  
01034  
01035  
01036  
01037  
01038  
01039  
01040  
01041  
01042  
01043  
01044  
01045  
01046  
01047  
01048  
01049  
01050  
01051  
01052  
01053  
01054  
01055  
01056  
01057  
01058  
01059  
01060  
01061  
01062  
01063  
01064  
01065  
01066  
01067  
01068  
01069  
01070  
01071  
01072  
01073  
01074  
01075  
01076  
01077  
01078  
01079  
01080  
01081  
01082  
01083  
01084  
01085  
01086  
01087  
01088  
01089  
01090  
01091  
01092  
01093  
01094  
01095  
01096  
01097  
01098  
01099  
01100  
01101  
01102  
01103  
01104  
01105  
01106  
01107  
01108  
01109  
01110  
01111  
01112  
01113  
01114  
01115  
01116  
01117  
01118  
01119  
01120  
01121  
01122  
01123  
01124  
01125  
01126  
01127  
01128  
01129  
01130  
01131  
01132  
01133  
01134  
01135  
01136  
01137  
01138  
01139  
01140  
01141  
01142  
01143  
01144  
01145  
01146  
01147  
01148  
01149  
01150  
01151  
01152  
01153  
01154  
01155  
01156  
01157  
01158  
01159  
01160  
01161  
01162  
01163  
01164  
01165  
01166  
01167  
01168  
01169  
01170  
01171  
01172  
01173  
01174  
01175  
01176  
01177  
01178  
01179  
01180  
01181  
01182  
01183  
01184  
01185  
01186  
01187  
01188  
01189  
01190  
01191  
01192  
01193  
01194  
01195  
01196  
01197  
01198  
01199  
01200  
01201  
01202  
01203  
01204  
01205  
01206  
01207  
01208  
01209  
01210  
01211  
01212  
01213  
01214  
01215  
01216  
01217  
01218  
01219  
01220  
01221  
01222  
01223  
01224  
01225  
01226  
01227  
01228  
01229  
01230  
01231  
01232  
01233  
01234  
01235  
01236  
01237  
01238  
01239  
01240  
01241  
01242  
01243  
01244  
01245  
01246  
01247  
01248  
01249  
01250  
01251  
01252  
01253  
01254  
01255  
01256  
01257  
01258  
01259  
01260  
01261  
01262  
01263  
01264  
01265  
01266  
01267  
01268  
01269  
01270  
01271  
01272  
01273  
01274  
01275  
01276  
01277  
01278  
01279  
01280  
01281  
01282  
01283  
01284  
01285  
01286  
01287  
01288  
01289  
01290  
01291  
01292  
01293  
01294  
01295  
01296  
01297  
01298  
01299  
01300  
01301  
01302  
01303  
01304  
01305  
01306  
01307  
01308  
01309  
01310  
01311  
01312  
01313  
01314  
01315  
01316  
01317  
01318  
01319  
01320  
01321  
01322  
01323  
01324  
01325  
01326  
01327  
01328  
01329  
01330  
01331  
01332  
01333  
01334  
01335  
01336  
01337  
01338  
01339  
01340  
01341  
01342  
01343  
01344  
01345  
01346  
01347  
01348  
01349  
01350  
01351  
01352  
01353  
01354  
01355  
01356  
01357  
01358  
01359  
01360  
01361  
01362  
01363  
01364  
01365  
01366  
01367  
01368  
01369  
01370  
01371  
01372  
01373  
01374  
01375  
01376  
01377  
01378  
01379  
01380  
01381  
01382  
01383  
01384  
01385  
01386  
01387  
01388  
01389  
01390  
01391  
01392  
01393  
01394  
01395  
01396  
01397  
01398  
01399  
01400  
01401  
01402  
01403  
01404  
01405  
01406  
01407  
01408  
01409  
01410  
01411  
01412  
01413  
01414  
01415  
01416  
01417  
01418  
01419  
01420  
01421  
01422  
01423  
01424  
01425  
01426  
01427  
01428  
01429  
01430  
01431  
01432  
01433  
01434  
01435  
01436  
01437  
01438  
01439  
01440  
01441  
01442  
01443  
01444  
01445  
01446  
01447  
01448  
01449  
01450  
01451  
01452  
01453  
01454  
01455  
01456  
01457  
01458  
01459  
01460  
01461  
01462  
01463  
01464  
01465  
01466  
01467  
01468  
01469  
01470  
01471  
01472  
01473  
01474  
01475  
01476  
01477  
01478  
01479  
01480  
01481  
01482  
01483  
01484  
01485  
01486  
01487  
01488  
01489  
01490  
01491  
01492  
01493  
01494  
01495  
01496  
01497  
01498  
01499  
01500  
01501  
01502  
01503  
01504  
01505  
01506  
01507  
01508  
01509  
01510  
01511  
01512  
01513  
01514  
01515  
01516  
01517  
01518  
01519  
01520  
01521  
01522  
01523  
01524  
01525  
01526  
01527  
01528  
01529  
01530  
01531  
01532  
01533  
01534  
01535  
01536  
01537  
01538  
01539  
01540  
01541  
01542  
01543  
01544  
01545  
01546  
01547  
01548  
01549  
01550  
01551  
01552  
01553  
01554  
01555  
01556  
01557  
01558  
01559  
01560  
01561  
01562  
01563  
01564  
01565  
01566  
01567  
01568  
01569  
01570  
01571  
01572  
01573  
01574  
01575  
01576  
01577  
01578  
01579  
01580  
01581  
01582  
01583  
01584  
01585  
01586  
01587  
01588  
01589  
01590  
01591  
01592  
01593  
01594  
01595  
01596  
01597

# FM-7 BIOSソース・リスト

```

00341      0006  A RS232D EQU $0006
00342      0007  A RS232S EQU $0007
00343      0007  A RS232C EQU $0007
00344      *
00345      000C  A ADCONV EQU $000C
00346      *
00347      0010  A BDATA EQU $0010
00348      0011  A BCMD EQU $0011
00349      0012  A BSTAT EQU $0012
00350      0013  A BERRST EQU $0013
00351      0014  A BPGADH EQU $0014
00352      0015  A BPGADL EQU $0015
00353      0016  A BPGGTH EQU $0016
00354      0017  A BPGCTL EQU $0017
00355      *
00356      *
00357      0018  A DSTAT EQU $18
00358      0018  A DCMD EQU $18
00359      0019  A DTRACK EQU $19
00360      001A  A DSECTR EQU $1A
00361      001B  A DDATA EQU $1B
00362      001C  A DSIDE EQU $1C
00363      001D  A DDRIVE EQU $1D
00364      001E  A DITREQ EQU $1E
00365      001F  A DDRQ EQU $1F
00366      *
00367      0020  A IANROA EQU $0020
00368      0022  A IANROD EQU $0022
00369      0021  A SIPI1 EQU $21
00370      002C  A SKIP2 EQU $2C
00371      002E  A SIPI2 EQU $2E
00372      1A50  A MSIFIO EQU $1A50
00373      000A  A FDFEAD EQU $000A
00374      0008  A FDFEST EQU $0008
00375      00FF  A IFEQ FMLCPH
00376      *
00377      *
00378      *

```

```

00379      0000  A IFEQ FBASIC
00380      *
00381      *
00382      *
00383      *
00384      *
00385      *
00386      *
00387      *
00388      *
00389      *
00390      *
00391      *
00392      *
00393      *
00394      *
00395      *
00396      *
00397      *
00398      *
00399      *
00400      *
00401      *
00402      *
00403      *
00404      *
00405      *
00406      *
00407P 0000  56  A FCC /V/
00408P 0001  01  A FCB $01
00409P 0002  4C  A FCC /L/
00410P 0003  01  A FCB $01
00411      *
00412P 0004  82  A FCB $82 YEAR
00413P 0005  09  A FCB $09 MONTH
00414P 0006  01  A FCB $01 DAY
00415      *

```

```

00417      0007  P CBIOS EQU *
00418P 0007  34  7F  A PSHS *
00419P 0008  86  FD  A LDA *
00420P 0008  1F  88  A TFR *
00421P 000D  E5  84  A LDB *
00422P 000F  4F  * CLRA
00423P 0010  58  * ASLB
00424P 0011  49  * ROLA
00425P 0012  108E 0033 P LDY #BIOSTB
00426P 0016  31  AB  P LEAY D,Y
00427P 0018  86  02  A LDA #2
00428P 001A  108C 0069 P CMPLY #BTEND
00429P 001E  22  06  0026 BHI BIOS1
00430P 0020  34  10  A PSHS X
00431P 0022  AD  B4  A JSR [0,Y]
00432P 0024  35  10  A PULS X
00433P 0026  36  01  A BIOS1 PULS
00434P 0028  A7  01  A STA CC
00435P 002A  26  03  002F BNE BIOS2
00436P 002C  1C  FE  A CLC
00437P 002E  8E  A FCB SKIP22
00438P 002F  1A  01  A BIOS2 SEC
00439P 0031  35  FE  A BIOS25 PULS
00441      0033  P BIOS2 EQU *
00442P 0033  0066 P ANALGP
00443P 0035  012D P MOTOR
00444P 0037  018A P CTEURT
00445P 0039  018A P CTERED
00446P 003B  0352 P INTEBL
00447P 003D  082C P SCREEN
00448P 003F  0383 P WRTBBL
00449P 0041  0383 P REDBBL
00450P 0043  FE02 A RESTOR
00451P 0045  FE05 A DWRITE
00452P 0047  FE08 A DREAD
00453P 0049  0432 P BEEP50
00454P 004B  0426 P BEEPON

```

```

00455P 004D  042E P FDB
00456P 004F  0434 P FDB
00457P 0051  0642 P FDB
00458P 0053  0478 P FDB
00459P 0055  0478 P FDB
00460P 0057  04DF P FDB
00461P 0059  06D0 P FDB
00462P 005B  06E7 P FDB
00463P 005D  0619 P FDB
00464P 005F  00A9 P FDB
00465P 0061  0451 P FDB
00466P 0063  098F P FDB
00467P 0065  0432 P FDB
00468P 0067  0432 P FDB
00469P 0069  0432 P BTEND
00471      *
00472      *
00473      *
00474P 006B A6 006B P ANALGP
00475P 006D 84 02 A LDA
00476P 006F 34 02 A ANDA
00477P 0071 E6 03 A PSHS
00478P 0073 4F 03 A LDB
00479P 0074 C1 48 A CLRA
00480P 0076 26 007A A CMPB
00481P 0078 86 04 A BNE
00482P 007A AA E4 A LDA
00483P 007C A7 E8 A RSSET
00484P 007E 8A 04 A STA
00485P 0080 37 0C A ORA
00486P 0082 84 07 A STA
00487P 0084 37 0C A ANDA
00488P 0086 34 02 A STA
00489P 0088 36 03 A PSHS
00490P 008A 8F E4 A LDA
00491P 008C 88 E4 A CLR
00492P 008E C6 80 A SDRDRA

```

```

00493P 0090 EA 61 A ORB
00494P 0092 D7 0C A STB
00495P 0094 C4 07 A ANDB
00496P 0096 D7 0C A STB
00497P 0098 D6 0C A LDB
00498P 009A C4 01 A ANDB
00499P 009C EB E4 A ADDB
00500P 009E E7 E4 A STB
00501P 00A0 4A 4A A DECA
00502P 00A1 26 E9 008C A BNE
00503P 00A3 35 06 A PULS
00504P 00A5 A7 04 A STA
00505P 00A7 4F 04 A CLRA
00506P 00A8 39 RTS
00508      *
00509      *
00510      *
00511P 00A8 EE 02 00A9 P KANJIR
00512P 00AB 108E 0000 A LDA
00513P 00AF C6 10 A LDY
00514P 00B1 10AF C1 A BFLCL
00515P 00B4 5A FA 00B1 A DECB
00516P 00B5 26 10 A BNE
00517P 00B7 C6 10 A LDB
00518P 00B9 34 04 A PSHS
00519P 00BB EC 04 A LOD
00520P 00BD 80 1C 000B A BSR
00521P 00BF 26 17 000D A SCS
00522P 00C1 EE 02 A LDY
00523P 00C3 D0 20 A LANEKD
00524P 00C5 108E 22 A LDY
00525P 00C8 108C 0001 A CMPLY
00526P 00CC 27 0A 00D8 A BEQ
00527P 00CE 10AF C1 A STY
00528P 00D1 C3 0001 A ADDO
00529P 00D4 6A E4 A DEC
00530P 00D6 26 EB 00C3 A BNE

```

```

00531P 00D8 4F 84 A KANRST
00532P 00D9 35 84 A CLRA
00533      *
00534      *
00535      *
00536      *
00537      *
00538      *
00539      *
00540      *
00541      *
00542      *
00543      *
00544      *
00545      *
00546      *
00547      *
00548      *
00549      *
00550      *
00551      *
00552      *
00553      *
00554      *
00555      *
00556P 00DB 84 7F A CNVJTK
00557P 00DD 81 21 A ANDA
00558P 00DF 25 10 00F1 A CMPA
00559P 00E1 81 28 A ELO
00560P 00E3 25 08 00ED A ELO
00561P 00E5 81 30 A CMPA
00562P 00E7 25 08 00F1 A ELO
00563P 00E9 81 50 A CMPA
00564P 00EB 24 04 00F1 A BHS
00565P 00ED C5 60 A SCDDC
00566P 00EF 26 03 00F4 A BNE
00567P 00F1 1A 01 NOROM

```

```

BEEPOF 13 BELL OFF
LPOUT 14 PRINTER OUTPUT
15 CRT HARD COPY
SUEOUT 16 SUBSYSTEM OUTPUT
SUBIN 17 SUBSYSTEM INPUT
INPUT 18 CONSOLE INPUT
INPUTC 19 CONSOLE INPUT CONTINUE
OUTPUT 20 CONSOLE OUTPUT
KEYIN 21 KEY INPUT
KANJIR 22 KANJI ROM DATA READ.
LPCHK 23 LP READY CHECK
BIINIT 24 BIOS INITIAL SET
BEEP50 25 RESERVED
BEEP50 26 RESERVED
BEEP50 27 RESERVED

```

```

*****
# ANALOG PORT INPUT ROUTINE.
RCBCNL,X GET CHANNEL NUMBER..
#03 EXTRACT C1,C0
A
RCBRNG,X GET VOLTAGE RANGE.

#*H HIGH VOLTAGE RANGE?
RSSET NO
#04 RS=1 SET
,S
,S
#08
ADCONV
#07
ADCONV
A
#3 COUNTER=9 SET.
,S WORK AREA=0 CLEAR
,S RESULT SHIFT LEFT 1 BIT.
#08

```

```

1,S
ADCONV
#07
ADCONV
ADCONV READ SERIAL DATA (LSB)
#01 EXTRACT SERIAL 1 BIT DATA.
0,S THIS TIME DATA ADD.
,S RESULT STORE.
COUNT DOWN.

SDREAD
0 GET RESULT ONE BYTE.
RCBAPD,X RESULT STORE
NORMAL RETURN

```

```

*****
# KANJI ROM READ ROUTINE
RCBDEA,X GET BUFFER ADDRESS
#0
#16 SET COUNTER
0,U++
32 BYTE 0 CLEAR
BFLCF
#16
B LOOP COUNTER SET.
RCBJCD,X GET THE KANJI JIS CODE.
CNVJJI CONVERT JIS CODE TO ROM ADDRESS
KANRTS NOT ON THE ROM.
RCBDEA,X GET KANJI BUFFER ADDRESS.
KANROA ADDRESS SET TO ROM ADDRESS REGISTER.
KANROD GET KANJI DATA
#1 NULL DATA ?
KANRTS NON DATA.
0,U++
#1 ADDRESS COUNTER UP
,S DECREMENT COUNTER BY 1.
KANEXO

```

```

NORMAL RETURN
E,PC
THE JIS-KANJI CODE TO THE ROM ADDRESS.
--KANJI CODE.
--ROM ADDRESS.
GISTER ARE RESERVED.

```

```

JIS(4:0)
-JIS(11:8)
04:12-2143+JIS(6:5)-11
WHEN T<-T-11
T=2 THEN DO:

#07F IS THE CODE ON THE KANJI ROM ?
#021
NOROM NO.
#028
SCDCK
#030
NOROM NO.
#050
NOROM NO.
#060
ONROM YES.

```

```

00668P 00F3 39 RTS
00669
00670P 00F4 34 06 A ONROM PSHS 0 SAVE THE JIS.
00671P 00FE 34 0F A ANDA #0F PICL UP JIS(11:8)
00672P 00F8 C4 1F A ANDOE PICL UP JIS(4:0)
00673P 00FA 58 LSLE SET IT TO A(12:4)
00674P 00FE 58 LSLE AND OTHER FIELDS IS SET 0.
00675P 00FC 58 LSLE
00676P 00FD 58 LSLE
00677P 00FE 49 FOLA
00678P 00FF 34 06 A PSHS 0
00679P 0101 A6 5C A LDA 2,S SAVE THE ADDRESS.
00680P 0103 84 70 A ANDA 2,S PICKUP JIS(14:12)
00681P 0105 80 20 A SUBA 2,S AND SUBTRACT 2 FROM IT.
00682P 0107 34 02 A PSHS A
00683P 0109 48 LSLE MULTIPLY IT BY 3.
00684P 010A A6 E4 A ADDA 3,S
00685P 010C 48 LSLE
00686P 010D A7 E4 A STA 3,S SET IT TO THE HIGHEST
00687P 010F A6 64 A LDA 4,S 3 BITS AND SAVE.
00688P 0111 84 60 A ANDA 4,S PICKUP JIS(6:5) AND
00689P 0113 80 20 A SUBA #00 SUBTRACT 1.
00690P 0115 A8 E4 A ADDA 3,S
00691P 0117 25 09 0122 BCS CNVJHI MAKE THE 1.
00692P 0119 81 40 A CMPA #2132
00693P 011B 25 07 0124 BLO CNVJLO IS THE T .LT. 2 ?
00694P 011D 22 03 0122 BHI CNVJHI YES.
00695P 011F 86 10 A LDA #19 NO,IT IS 2.
00696P 0121 8C 8C A FCS NO,IT IS 2. SET A(12) BIT
00697P 0122 80 20 A CNVJHI SUBA #11432 SHIP NEXT 2-BYTES (CMPX #)
00698P 0124 A4 61 A CNVJLO ORA 1,S SUBTRACT 1 FROM THE T.
00699P 0126 E6 62 A LDB 2,S SET THE T TO A(15:13).
00700P 0128 32 65 A LEAS 5,S LOAD THE ADDRESS TO D-REG.
00701P 012A 1C FE CLC
00702P 012C 39 RTS
00703 0042 A MOTRON EQU #42
00704 0040 A MOTROF EQU #40

```

```

00606 *****
00607
00608 012D P MOTOP EQU 1 CASSETTE MOTOR CONTROL.
00609P 012D A6 02 A LDA MOTOF,X GET MOTOR FLAG.
00610P 012F C6 42 A LDB #MOTRON ASSUME MOTOR ON.
00611P 0131 43 COMA MOTOR ON?
00612P 0132 27 02 0136 BEQ MOTRON YES,MOTOR ON.
00613P 0134 C6 40 A LDB #MOTROF NO,MOTOR OFF.
00614P 0136 D7 00 A MOTRON STB REMOTE
00615P 0138 4F CLRA MOTOR CONTROL SET.
00616P 0139 39 RTS NORMAL RETURN.
00617
00618 *****
00619
00620
00621 013A P CTWRT EQU 1
00622P 013A 96 00 A LDA MTLPT
00623P 013C 44 LSRA 2 MHZ ?
00624P 013D 1025 016D 024E LBSC 22MHZW YES 2 MHZ
00625P 0141 A6 02 A LDA CRWDAT,X GET WRITE DATA.
00626P 0143 8D 63 01A8 BSR DOUT
00627P 0145 8E 0006 A LDX #6
00628P 0148 8D 68 01B2 BSR TIMER
00629P 014A 8D 61 01AD W20 BSR D1OUT
00630P 014C 8E 0012 A LDX #18
00631P 014F 8D 61 01B2 BSR TIMER
00632P 0151 8D 55 01A8 W30 BSR DOUT
00633P 0153 8E 0012 A LDX #18
00634P 0156 8D 5A 01B2 BSR TIMER
00635P 0158 108E 0000 A LDY #0
00636P 015C 8D 4F 01AD W40 BSR D1OUT
00637P 015E 44 LSRA
00638P 015F 25 0E 016F BCS WRITE1
00639P 0161 8E 0011 A WRITE0 LDX #17
00640P 0164 8D 4C 01B2 BSR TIMER
00641P 0166 8D 40 01A8 W50 BSR DOUT
00642P 0168 8E 0011 A LDX #17
00643P 016E 8D 45 01B2 BSR TIMER

```

```

00644P 016D 20 0E 017D BRA NEXT
00645P 016F 8E 0024 A WRITE1 LDX #36
00646P 0172 8D 3E 01B2 BSR TIMER
00647P 0174 8D 32 01A8 W50 BSR DOUT
00648P 0176 8E 0023 A LDX #36
00649P 0178 8D 37 01B2 BSR TIMER
00650P 017B 20 00 017D BRA NEXT
00651P 017D 31 21 A LEAY 1,Y
00652P 017F 108C 0008 A CMPY #8
00653P 0183 26 07 015C BNE W40
00654P 0185 108E 0000 A LDY #0
00655P 0189 8D 22 01AD W70 BSR D1OUT
00656P 018B 8E 0024 A LDX #36
00657P 018E 8D 22 01B2 BSR TIMER
00658P 0190 8D 16 01A8 W80 BSR DOUT
00659P 0192 8E 0010 A LDX #16
00660P 0195 8D 18 01B2 BSR TIMER
00661P 0197 31 21 A LEAY 1,Y
00662P 0199 108C 0002 A CMPY #2
00663P 019D 27 07 01A6 BEQ BYTEND
00664P 019F 8E 0016 A LDX #22
00665P 01A2 8D 0E 01B2 BSR TIMER
00666P 01A4 20 E3 0189 BFA W70
00667P 01A6 4F BYTEND CLRA
00668P 01A7 39 RTS
00669
00670P 01A8 C6 42 A DOUT LDB #042
00671P 01AA D7 00 A STB WRITE
00672P 01AC 39 RTS
00673
00674P 01AD C6 43 A D1OUT LDB #043
00675P 01AF D7 00 A STB WRITE
00676P 01B1 39 RTS DATA OUT.
00677
00678P 01B2 30 1F A TIMER LEAX -1,X
00679P 01B4 8C 0000 A CMPX #0
00680P 01B7 26 F9 01B2 BNE TIMER

```

```

00644P 016D 20 0E 017D BRA NEXT
00645P 016F 8E 0024 A WRITE1 LDX #36
00646P 0172 8D 3E 01B2 BSR TIMER
00647P 0174 8D 32 01A8 W50 BSR DOUT
00648P 0176 8E 0023 A LDX #36
00649P 0178 8D 37 01B2 BSR TIMER
00650P 017B 20 00 017D BRA NEXT
00651P 017D 31 21 A LEAY 1,Y
00652P 017F 108C 0008 A CMPY #8
00653P 0183 26 07 015C BNE W40
00654P 0185 108E 0000 A LDY #0
00655P 0189 8D 22 01AD W70 BSR D1OUT
00656P 018B 8E 0024 A LDX #36
00657P 018E 8D 22 01B2 BSR TIMER
00658P 0190 8D 16 01A8 W80 BSR DOUT
00659P 0192 8E 0010 A LDX #16
00660P 0195 8D 18 01B2 BSR TIMER
00661P 0197 31 21 A LEAY 1,Y
00662P 0199 108C 0002 A CMPY #2
00663P 019D 27 07 01A6 BEQ BYTEND
00664P 019F 8E 0016 A LDX #22
00665P 01A2 8D 0E 01B2 BSR TIMER
00666P 01A4 20 E3 0189 BFA W70
00667P 01A6 4F BYTEND CLRA
00668P 01A7 39 RTS
00669
00670P 01A8 C6 42 A DOUT LDB #042
00671P 01AA D7 00 A STB WRITE
00672P 01AC 39 RTS
00673
00674P 01AD C6 43 A D1OUT LDB #043
00675P 01AF D7 00 A STB WRITE
00676P 01B1 39 RTS DATA OUT.
00677
00678P 01B2 30 1F A TIMER LEAX -1,X
00679P 01B4 8C 0000 A CMPX #0
00680P 01B7 26 F9 01B2 BNE TIMER

```

# FM-7 BIOSソースリスト

```

00681P 01B9 39          RTS
00682                *
00683                *
00684                *
00685                *
00686                *
00687P 01BA 34          01BA P CTBRED EQU # X CASSETTE BYTE READ.
00688P 01BC 36          00 A RCE ADDR SAVE.
00689P 01BE 44          00 A MTLPT
00690P 01BF 1025 0104 02C7 2 MHZ ?
00691P 01C3 108E 0000 A CBINO LDCS ZMMHCF YES 2 MHZ
00692                * COUNTER CLEAR.
00693                *
00694                *
00695                *
00696                *
00697                *
00698P 01C7 7D          0000 A IFEQ FM_CPM ASSEMBLE WHEN FM_CPM=0.
00699P 01CA 1026 FE32 0000 ENDC
00700                *
00701                *
00702P 01CE 8D          5A 0224 BSR DIN
00703P 01D0 26          F5 01C7 BNE R10 IF NOT,READ AGAIN.
00704P 01D2 80          52 0236 R20 BSR EDGE EDGE CATCH.
00705P 01D4 8E          001E A LDJ #30 TIMER SET.
00706P 01D6 80          D3 01E2 BSR TIMER
00707P 01D8 8D          4F 022A R30 BSR DIN
00708P 01DA 26          EA 01C7 BNE R10 IF IT IS NOT CAUGHT,START BIT AGAIN.
00709P 01DC 4F          0000 A CLR A
00710P 01DE 80          5A 0236 R40 BSR EDGE EDGE CATCH.
00711P 01E0 8E          001B A LDJ #27 TIMER SET.
00712P 01E2 8D          CD 01E2 BSR TIMER
00713P 01E4 80          43 022A R50 BSR DIN
00714P 01E6 8D          0000 A LSLB
00715P 01E8 46          0000 A RORA
00716P 01EA 31          21 A LEAY 1,Y
00717P 01EC 31          21 A COUNTER INC.
00718P 01EE 108C 0008 A CMPLY #8 AFE DATA 8 BIT ?
00719P 01EF 27          0F 0200 BEQ STOP IF OK,GO TO STOP BIT CATCH.
    
```

```

00721                *
00722                *
00723                *
00724                *
00725                *
00726P 01F1 7D          0000 A IFEQ FM_CPM ASSEMBLE WHEN FM_CPM=0.
00727P 01F4 1026 FE08 0000 ENDC
00728                *
00729                *
00730P 01F8 D6          02 A LDE READ DATA IN.
00731P 01FA C4          80 A ANDB #80
00732P 01FC 26          F3 01F1 BNE R60 IF NOT,READ AGAIN.
00733P 01FE 20          DE 01DE BRA R40 GO TO NEXT BIT READ.
00734                *
00735                *
00736P 0200 108E 0000 A STOP LDY #0 COUNTER CLEAR.
    
```

```

00738                *
00739                *
00740                *
00741                *
00742                *
00743P 0204 7D          0000 A IFEQ FM_CPM ASSEMBLE WHEN FM_CPM=0.
00744P 0207 1026 FDF5 0000 ENDC
00745                *
00746                *
00747P 0208 D6          02 A LDB READ DATA IN.
00748P 020A C4          80 A ANDB #80
00749P 020C 26          F3 0204 BNE R70 IF NOT,READ AGAIN.
00750P 020E 8D          23 0236 R80 BSR EDGE EDGE CATCH.
00751P 0210 80          0018 A LDJ #24
00752P 0212 8E          9A 01E2 BSR TIMER
00753P 0214 8D          10 022A R90 BSR DIN
00754P 0216 80          A7 01C3 BEQ CBINO
00755P 0218 27          21 A LEAY 1,Y
00756P 021A 31          21 A COUNTER INC.
00757P 021C 108C 0002 A CMPLY #2 ARE 2 STOP BIT CAUGHT ?
00758P 021E 26          E0 0204 BNE R70 GO TO 2ND STOP BIT.
    
```

```

00759P 0224 35          10 A PULS X PCB COME BACK.
00760P 0226 A7          02 A STA CRWDAT,X
00761P 0228 4F          80 A CLRA
00762P 0229 39          RTS
00763                *
00764P 022A D6          02 A DIN LDB READ DATA IN.
00765P 022C C4          80 A ANDB #80
00766P 022E 39          RTS
00767                *
00768P 022F 35          10 A ERROR PULS X
00769P 0231 5F          02 A CLR CRWDAT,X
00770P 0233 86          34 A LDA #52 ABNORMAL READ
00771P 0235 39          RTS
    
```

```

00772                *
00773                *
00774                *
00775                *
00776                *
00777                *
00778                *
00779P 0236 7D          0000 A IFEQ FM_CPM ASSEMBLE WHEN FM_CPM=0.
00780P 0239 1026 FDC3 0000 ENDC
00781                *
00782                *
00783P 023D D6          02 A LDB READ DATA IN.
00784P 023F C4          80 A ANDB #80
00785P 0241 C1          80 A CMPEB #80
00786P 0243 26          F1 0236 BNE R70 IF NOT,READ AGAIN.
00787P 0245 D6          02 A LDB READ DATA IN.
00788P 0247 C4          80 A ANDB #80
00789P 0249 C1          80 A CMPEB #80
00790P 024B 26          E3 0236 BNE R70 IF NOT,READ AGAIN.
00791P 024D 39          RTS
00792P 024D 39          E3 0236 BNE R70
00793                *
00794                *
00795P 024E A6          02 A LDA CRWDAT,X GET WRITE DATA
00796P 0250 8D          63 02B5 BSR ZDOUT
00797P 0252 8E          000B A LDJ #645
00798P 0254 8D          68 02BF BSR ZTIMER
    
```

```

00799P 0257 8D          61 02BA ZDOUT BSR
00800P 0259 8E          0020 A LDJ
00801P 025C 8D          61 02BF BSR
00802P 025E 8D          55 02B5 ZDOUT3 BSR
00803P 0260 8E          0020 A LDJ
00804P 0263 8D          5A 02BF BSR
00805P 0265 108E 0000 A LDJ
00806P 0268 8D          4F 02BA ZDOUT4 BSR
00807P 026A 44          0E 027C LDB
00808P 026C 25          001E A ZDOUT1 BSR
00809P 026E 8E          4C 02BF BSR
00810P 0271 8D          43 02B5 ZDOUT5 BSR
00811P 0273 8D          001F A LDJ
00812P 0275 8E          45 02BF BSR
00813P 0278 8D          0E 02BA BSR
00814P 027A 20          003E A ZDOUT1 BSR
00815P 027C 8E          3E 02BF BSR
00816P 027F 8D          32 02B5 ZDOUT6 BSR
00817P 0281 8D          003D A LDJ
00818P 0283 8E          37 02BF BSR
00819P 0286 8D          00 02BA BSR
00820P 0288 20          21 A ZNEXT
00821P 028A 31          21 A LEAY
00822P 028C 108C 0008 A CMPLY
00823P 028E 26          D7 02B8 BNE
00824P 0292 108E 0000 A LDJ
00825P 0296 8D          22 02BA ZDOUT7 BSR
00826P 0298 8E          003D A LDJ
00827P 029B 8D          22 02BF BSR
00828P 029D 8D          16 02B5 ZDOUT8 BSR
00829P 029F 8E          0020 A LDJ
00830P 02A2 8D          1B 02BF BSR
00831P 02A4 31          21 A LEAY
00832P 02A6 108C 0002 A CMPLY
00833P 02AA 27          07 02B3 BSR
00834P 02AC 8E          0025 A LDJ
00835P 02AF 8D          0E 02BF BSR
    
```

```

00836P 02B1 20          E3 02B6 BRA
00837P 02B3 4F          00 ZBYTEN CLRA
00838P 02B4 39          RTS
00839                *
00840P 02B5 D6          42 A ZDOUT0 LDB
00841P 02B7 D7          00 A STB
00842P 02B9 39          RTS
00843                *
00844P 02BA D6          43 A ZDOUT1 LDB
00845P 02BC D7          00 A STB
00846P 02BE 39          RTS
00847                *
00848P 02BF 30          1F A ZTIMER LEAY
00849P 02C1 8C          0000 A CMPLY
00850P 02C4 26          F9 02BF BNE
00851P 02C6 39          RTS
    
```

```

00852                *
00853                *
00854P 02C7 108E 0000 P ZMMH2R
00855P 02C9 0000 A ZCBINO LDY
00856                *
00857                *
00858                *
00859                *
00860P 02CB 7D          0000 A IFEQ
00861P 02CE 1026 FDE2 0000 ENDC
00862                *
00863                *
00864P 02D2 8D          5A 032E BSR
00865P 02D4 26          F5 02CB BNE
00866P 02D6 8D          62 033A ZR20 BSR
00867P 02D8 8E          003E A LDJ
00868P 02DA 8D          E2 02BF BSR
00869P 02DC 8D          4F 032E ZR30 BSR
00870P 02DE 26          EA 02CB BNE
00871P 02E1 4F          56 033A ZR40 CLRA
00872P 02E2 31          0030 A LDJ
00873P 02E4 8E          0030 A ANDB
00874P 02E7 8D          D5 02BF BSR
00875P 02E9 8D          43 032E ZR50 BSR
    
```

```

00876P 02EB 58          0000 A LSLE
00877P 02ED 46          21 A RORA
00878P 02ED 31          21 A LEAY
00879P 02EF 108C 0008 A CMPLY
00880P 02F3 27          0F 0304 BEQ
00881                *
00882                *
00883                *
00884                *
00885                *
00886P 02F5 7D          0000 A IFEQ
00887P 02F8 1026 FDF4 0000 ENDC
00888                *
00889P 02FC D6          02 A LDB
00890P 02FE C4          80 A ANDB
00891P 0300 26          F3 02F5 BNE
00892P 0302 20          DE 02E2 BRA
00893                *
00894                *
00895P 0304 108E 0000 A ZSTOP LDY
00896P 0306 0000 A IFEQ
00897                *
00898                *
00899                *
00900                *
00901P 0308 7D          0000 A IFEQ
00902P 030B 1026 FCF1 0000 ENDC
00903                *
00904P 030F D6          02 A LDB
00905P 0311 C4          80 A ANDB
00906P 0313 26          F3 0308 BNE
00907P 0315 8D          23 033A ZR80 BSR
00908P 0317 8E          002C A LDJ
00909P 031A 8D          A3 02BF BSR
00910P 031C 8D          10 032E ZR90 BSR
00911P 031E 27          A7 02C7 BEQ
00912P 0320 31          21 A LEAY
00913P 0322 108C 0002 A CMPLY
00914P 0324 26          E0 0308 BNE
00915P 0326 35          10 A PULS
    
```



```

ZD10UT      TIMER SET
#18+14
ZTIMER
ZD00OUT
#18+14      TIMER SET
ZTIMER
#0          COUNTER CLEAR
ZD10UT
          IS WRITING DATA 0 2, OR 1 ?
          IF IT IS 1, GOTO Z2WRT1
#17+13      TIMER SET
ZTIMER
ZD00OUT
#17+14      TIMER SET
ZTIMER
ZNEXT
#36+26      TIMER SET
ZTIMER
ZD00OUT
#35+26      TIMER SET
ZTIMER
ZNEXT
1,Y        BIT COUNTS
#8          1 BYTE END ?
Z2W40      IF NOT, GOTO NEXT BIT
#0          COUNTER CLEAR
ZD10UT
#36+26      TIMER SET
ZTIMER
ZD00OUT
#12+20      TIMER SET
ZTIMER
1,Y        COUNTER INC.
#2          IS 2 STOP BITS WRITTEN ?
ZBYTEN     IF YES, GOTO 1 BYTE END
#22+15     TIMER SET
ZTIMER

```

```

Z2W70      GOTO 2ND STOP BIT
          NORMAL RETURN
#42        OUTPUT LOW LEVEL
WRITE      DATA OUT
#43        OUTPUT HIGH-LEVEL
WRITE      DATA OUT
-1,X
#0          ZTIMER
#0          COUNTER CLEAR
FM_CPM     ASSEMBLE WHEN FM_CPM=0
FBASIC     ASSEMBLE WHEN FBASIC=0
ABORTF     STOP KEY ON ?
ZDIN
ZK10       IF NOT, READ AGAIN
ZEDGE      EDGE CATCH
#30+24     TIMER SET
ZTIMER
ZDIN
ZK10       IF IT IS NOT CAUGHT, START BIT AGAIN
ZEDGE      EDGE CATCH
#25+23     ZTIMER
ZDIN

```

```

1,Y        SET DATA IN A-REG
#3          COUNTER INC.
#3          ARE DATA 8 BIT ?
ZSTOP      IF OK, GOTO STOP BIT CATCH
FM_CPM     ASSEMBLE WHEN FM_CPM=0

```

```

FBASIC     ASSEMBLE WHEN FBASIC=0
ABORTF     STOP KEY ON ?
ABTCHK

```

```

READ      DATA IN
#80
ZK70      IF NOT, READ AGAIN
ZK40      GOTO NEXT BIT READ

```

```

#0          COUNTER CLEAR
FM_CPM     ASSEMBLE WHEN FM_CPM=0

```

```

FBASIC     ASSEMBLE WHEN FBASIC=0
ABORTF     STOP KEY ON ?
ABTCHK

```

```

READ      DATA IN
#80
ZK70      IF NOT, READ AGAIN
ZEDGE      EDGE CATCH
#24+20     ZTIMER
ZDIN
ZCBINO
1,Y        COUNTER INC.
#2          ARE 2 STOP BIT CAUGHT ?
ZK70      GOTO 2-ND STOP BIT
X          RCB COME BACK

```

```

00817P 032A A7 02 A STA CRWDAT,X
00818P 032C 4F CLRA NORMAL READ
00819P 032D 39 RTS 1 BYTE DATA HAVE BEEN COUGHT
00820P
00821P 032E D6 02 A ZDIN LDB READ DATA IN
00822P 0330 C4 80 A ANDB #80
00823P 0332 39 RTS
00824P
00825P 0333 35 10 A ZERROR PULS X
00826P 0335 6F 02 A CLP CRWDAT,X
00827P 0337 56 34 A LDA #52 ABNORMAL READ
00828P 0339 39 RTS
00830P 00FF A IFEQ FM_CPM ASSEMBLE WHEN FM_CPM=0
00831P
00832P ENDC
00833P
00834P 033A 7D 0000 A IFEQ FBASIC ASSEMBLE WHEN FBASIC=0
00835P 033D 0026 FCBF 0000 ZEDGE ABORTF STOP KEY ON ?
00836P
00838P 0341 D6 02 A LDB READ DATA IN
00839P 0343 C4 80 A ANDB #80
00840P 0345 C1 80 A CMPB #80 RAISING EDGE OK ?
00841P 0347 26 F1 033A BNE ZEDGE IF NOT, READ AGAIN
00842P 0349 D6 02 A LDB READ DATA IN
00843P 034B C4 80 A ANDB #80 RAISING EDGE OK ?
00844P 034D C1 80 A CMPB #80
00845P 034F 26 E9 033A BNE ZEDGE IF NOT, READ AGAIN
00846P 0351 39 RTS
00848P
00849P
00850P 0352 0352 P INTBBL EQU # BUBBLE INITIALIZE
00851P 0352 1A50 A FDB MSKF10
00852P 0354 8D 3C 0392 BSR BUBRDY
00853P 0356 86 04 A LDA #504 INITIALIZE BUBBLE COMMAND SET
00854P 0358 97 11 A INTB05 STA BCDM
00855P 035A 96 12 A INTB10 BSTAT CME ?
00856P 035C 2A FC 035A EPL INTB10 NO, LOOP

```

```

00857P 035E 36 12 A LDA ESTAT CME ?
00858P 0360 2A F8 035A EPL INTB10 NO, LOOP
00859P 0362 85 02 A BITA #502
00860P 0364 26 02 0368 BNE BERR0P ERROR ANALYSIS
00861P 0366 4F 39 CLRA
00862P 0367 3F RTS
00863P
00864P 0368 P BERR0P EQU # ERROR ANALYSIS
00865P 0368 D6 13 A LDB BERRST LOAD ERROR STATUS REGISTER.
00866P 036A 85 04 A BITA #504 WRITE PROTECT ?
00867P 036C 26 21 036F BNE WRTPR YES WRITE PROTECT.
00868P 036E 85 10 A BITA #510 DEVICE NOT READY ?
00869P 0370 26 19 0368 BNE NONSUB YES BUBBLE NOT READY.
00870P 0372 86 1E A LDA #30 30
00871P 0374 54 1A 0391 LSRB UNDEFINED ?
00872P 0376 25 1A 0391 BCS BERR50
00873P 0377 4C 1A 0391 INCA BERR50
00874P 0378 54 16 0391 LSRB NO MARKER ?
00875P 0379 25 16 0391 BCS BERR50
00876P 037B 4C 1A 0391 INCA BERR50
00877P 037C 54 12 0391 LSRB MANY BAD LOOP ?
00878P 037D 25 12 0391 BCS BERR50
00879P 037F 4C 1A 0391 INCA BERR50
00880P 0380 54 0E 0391 LSRB TRANSFER MISSING ?
00881P 0381 25 0E 0391 BCS BERR50
00882P 0383 4C 0A 0391 INCA BERR50
00883P 0384 54 0A 0391 LSRB CRC ERROR ?
00884P 0385 25 0A 0391 BCS BERR50
00885P 0387 4C 1A 0391 INCA BERR50
00886P 0388 54 1A 0391 LSRB PAGE ADDRESS ERROR ?
00887P 0389 25 06 0391 BCS BERR50
00888P 038B 86 25 A NONSUB LDA #37 BUBBLE DEVICE NOT READY.
00889P 038D 20 02 0391 BNE BERR50
00890P
00891P 038F 86 24 A WRTPR LDA #36 WRITE PROTECT ERROR
00892P 0391 39 BERR50 RTS
00893P

```

```

00894P 0392 P BUBRDY EQU # BUBBLE REGISTER CHECK
00895P
00896P 0392 36 55 A LDA #55 $55 PATARN WRITE.
00897P 0394 37 10 A STA BDATA
00898P 0396 31 10 A BDATA EQUAL ?
00899P 0398 26 15 03AF BNE BUBERR BUBBLE INVALID,
01000P 039A 43 COMA $AA PATARN WRITE.
01001P 039B 37 10 A STA BDATA
01002P 039D 31 10 A BDATA EQUAL ?
01003P 039F 26 0E 03AF BNE BUBERR BUBBLE INVALID.
01004P # BUBBLE REGISTER IS VALID.
01005P 03A1 108E FFFF A LDY #FFFF
01006P 03A5 31 3F A BUBLOP LEAY -1,Y
01007P 03A7 27 06 03AF BEQ BUBERR
01008P 03A9 36 12 A LDA BSTAT
01009P 03AB 44 F7 03A5 LSRB BUBLOP
01010P 03AC 25 BCS
01011P 03AE 39 RTS
01012P
01013P 03AF 32 62 A BUBERR LEAS 2,S STACK ADJUSTMENT.
01014P 03B1 20 D8 038B ERA NONEUB
01015P 0007 A FEDEB EQU 7
01017P
01018P
01019P 03B3 03B3 P REDDBL EQU #
01020P 03B3 1A50 A WRTEBL MSHF10 IRQ MASK SET.
01021P 03B5 8D 08 0392 BSR BUBRDY
01022P 03B7 EC D6 A LDD RCBPCH,X
01023P 03B9 D0 16 A STD BPGRTH
01024P 03BB EC 04 A LDD RCBPAH,X
01025P 03BD 8D 27 03B6 BSR REALAD REAL ADDRESS SET. ADDRESS CHECK.
01026P 03BF D0 14 A STD BPGRADH
01027P 03C1 10AE 02 A LDY RCBDBA,X BUFFER ADDRESS
01028P 03C4 A6 84 A LDA RQNO,X
01029P 03C6 31 07 A CMPA #REDB REDDBL ?
01030P 03C8 27 49 0413 BEQ REDEBX BRANCH BUBBLE READ.
01031P 03CA 36 02 A LDA #502 COMMAND SET

```

# FM-7 BIOSソースリスト

```

01032P 03CC 37 11 A STA BCMD.
01033P 03CE 36 12 A LDA BSTAT
01034P 03D0 28 0A 03DC WRTB20 COMMAND END
01035P 03D2 85 40 A BITA #540
01036P 03D4 27 F8 03CE BEQ WRTB10 TDRA
01037P 03D6 A6 10 A LDA O,Y+
01038P 03D8 37 10 A STA BDATA
01039P 03DA 20 F2 03CE BRA WRTB10
01040
01041P 03DC 36 12 A WRTB20 LDA BSTAT
01042P 03DE 2A EE 03CE BPL WRTB10
01043P 03E0 85 02 A WRTB30 BITA #02
01044P 03E2 26 84 03E8 BNE BEFROR
01045P 03E4 4F A CLRFA
01046P 03E5 39 A RTS
01047
01048
01048 P REALAD EQU # ADDRESS,COUNTS CHECK.
01049 # UNIT OFFSET ADD.
01050P 03E6 34 06 A PSHS D
01051P 03E8 34 7F A ANDA #57F D-SAVE.
01052P 03EA 1083 03FF A CMPO #53FF B15 - 0 D:CASSETTE ADDRESS.
01053P 03EE 22 1E 040E BHI ADRERR NO,PAGE ADDRESS ERROR.
01054
01055P 03F0 EE 06 A LDU RCBPCH,X
01056P 03F2 1183 0400 A CMPO #5400 COUNTS <= $400 ?
01057P 03F6 22 16 040E BHI ADRERR NO,PAGE COUNTS ERROR.
01058
01058P 03F8 34 40 A PSHS U
01059P 03FA E3 E1 A ADD S,++ ++: STACK ADJUST.
01060P 03FC 1083 0400 A CMPO #5400 ADDR+COUNTS < $400 ?
01061P 0402 22 0C 040E BHI ADRERR NO,PAGE ADDRESS ERROR.
01062P 0404 26 0A 040E BPL D D-COME BACK.
01063P 0406 26 80 A BITA #80 UNIT 0 ?
01064P 0408 27 05 0400 BEQ RADRTS YES,UNIT 0
01065P 040A 34 7F A ANDA #57F ACCA: MSB=0
01066P 040C 03 0400 A ADDD #5400 NO,UNIT1, ADD UNIT0 OFFSET.
01067P 040E 39 A RADRTS RTS
01068

```

```

01070P 040E 86 23 A ADRERR LDA #35 PAGE ADDRESS ERROR.
01071P 0410 35 60 A PULS U,Y U:DUMMY FOR STACK ADJUST.
01072
01072P 0412 39 # RTS Y:FOR BSR REALAD
01073
01073P 0412 39 # RTS
01074
01074P 0412 39 # RTS
01075
01075P 0412 39 # RTS
01076
01076P 0412 39 # RTS
01077
01077P 0412 39 # RTS
01078P 0413 86 01 A REDBEX EQU #01 BUBBLE READ
01079P 0415 37 11 A LDA COMMAND SET
01080P 0417 36 12 A STA BCMD
01081P 0419 28 0A 0425 WRTB10 BSTAT
01082P 041B 85 20 A 0425 BMT REDB20 COMMAND END
01083P 041D 27 F8 0417 BEQ REDB10 RDA
01084P 041F 36 10 A LDA BDATA
01085P 0421 A7 10 A STA O,Y+
01086P 0423 20 F2 0417 BRA REDB10
01087
01088P 0425 36 12 A REDB20 EQU # REDB20
01089P 0427 2A EE 0417 LDA BSTAT
01090P 0429 20 85 03E0 BPL REDB10
01091
01091P 0429 20 85 03E0 BPL WRTB30
01092
01092P 0429 20 85 03E0 BPL WRTB30
01093
01093P 0429 20 85 03E0 BPL WRTB30
01094
01094P 0429 20 85 03E0 BPL WRTB30
01095
01095P 0429 20 85 03E0 BPL WRTB30
01096
01096P 0429 20 85 03E0 BPL WRTB30
01097
01097P 0429 20 85 03E0 BPL WRTB30
01098
01098P 0429 20 85 03E0 BPL WRTB30
01099
01099P 0429 20 85 03E0 BPL WRTB30
01100
01100P 0429 20 85 03E0 BPL WRTB30
01101
01101P 0429 20 85 03E0 BPL WRTB30
01102P 042E 86 01 A BEEPON EQU #01 BELL ON
01103P 0430 37 03 A LDA LQA
01104P 0432 4F A STA BELL
01105P 0434 39 A RTS
01106
01106P 0434 39 A RTS
01107
01107P 0434 39 A RTS
01108
01108P 0434 39 A RTS

```

```

01109
01109P 0434 10A6 02 A P LPOUT EQU # PRINTER OUTPUT
01110P 0436 EE 04 A LDY RCBDBA,X DATA BUFFER ADDRESS
01111P 0438 27 14 044F LDU RCBLNH,X OUTPUT BYTES NO.
01112P 0439 27 0A 044F BEQ LPOUT2 IF BYTES NO.=0 DO NOTHING
01113
01113P 0439 27 0A 044F P LPOUT1 EQU #
01114P 043B A6 A0 A LDA O,Y+
01115P 043D BD 07B2 P JSR LPPUTX
01116
01116P 043D BD 07B2 P JSR LPPUTX
01117
01117P 043D BD 07B2 P JSR LPPUTX
01118P 0440 7D 0313 A IFEQ FBASIC ASSEMBLE WHEN FBASIC=0.
01119P 0443 1026 FB69 0000 A TST ABORTF STOP KEY ON ?
01120
01120P 0443 1026 FB69 0000 A LBNX ENDC
01121P 0447 33 5F A LEAU -1,U ALL END ?
01122P 0449 1183 0000 A CMPO #0
01123P 044D 26 EC 0438 BNE LPOUT1
01124P 044F 4F A LPOUT2 CLRFA
01125P 0450 39 A RTS
01126
01126P 0450 39 A RTS
01127
01127P 0450 39 A RTS
01128
01128P 0450 39 A RTS
01129
01129P 0450 39 A RTS
01130
01130P 0451 D6 0451 P LPCHK EQU # PRINTER OUTPUT
01131P 0453 C5 02 A LDY MTLPIN ERROR ?
01132P 0455 02 02 A BITB #02
01133P 0457 27 1B 0472 BEQ LPCH40 YES ERROR
01134
01134P 0457 CE 0004 A LDU #4
01135P 045A 108E F000 A LPCH10 LDY #5F000 ABOUT 2.0 SECOND
01136P 045E 31 3F A LPCH20 LEAY -1,Y
01137P 0460 27 07 0469 BEQ LPCH30
01138P 0462 D6 02 A LDY MTLPIN
01139P 0464 54 A LSRB
01140P 0466 25 F7 045E BCS LPCH20 BUSY ?
01141P 0468 39 A CLRFA
01142P 046A 39 A RTS
01143P 046B 39 A RTS
01144
01144P 046B 39 A RTS
01145P 046B 39 A RTS
01146P 046B 1183 0000 A CMPO #0
01147P 046F 26 E9 045A BNE LPCH10
01148P 0471 5F A CLRFB

```

```

01149P 0472 86 32 A LPCH40 LDA
01150P 0474 C5 08 A BITB
01151P 0476 26 02 047A BNE
01152P 0478 86 33 A LDA
01153P 047A 39 A LPCH50 RTS
01154
01154P 047A 39 A LPCH50 RTS
01155
01155P 047A 39 A LPCH50 RTS
01156
01156P 047A 39 A LPCH50 RTS
01157
01157P 047A 39 A LPCH50 RTS
01158
01158P 047A 39 A LPCH50 RTS
01159P 047B 86 047B P SUBOUT EQU
01160P 047D 80 04 A EQU
01161P 047F 26 50 040E LDY
01162P 0481 E6 05 A ENE
01163P 0483 27 53 040E BEQ
01164P 0485 C1 80 A CMPB
01165P 0487 22 55 040E BHI
01166P 0489 34 10 A PSHS
01167P 048B 108E FC80 A LDY
01168P 048F AE 02 A LDX
01169P 0491 ED 0723 P JSR
01170P 0493 AE 01 A LDA
01171P 0495 84 80 A STA
01172P 0497 A7 01 A ANDA
01173P 0499 A6 21 A LDA
01174P 049B 84 7F A ANDA
01175P 049D AA 01 A ORA
01176P 049F A7 01 A STA
01177P 04A1 A6 80 A SUB20 LDA
01178P 04A3 A7 A0 A SUB25 STA
01179P 04A5 A6 A0 A DECE
01180P 04A7 26 F9 04A2 BNE
01181P 04A9 BD 073A P JSR
01182P 04AB 35 10 A PULS
01183P 04AD 4F A CLRFB
01184P 04AF E6 84 A LDA
01185P 04B1 C1 11 A CLRB
01186P 04B3 26 29 040E BNE

```

```

01187P 04B5 BD 071E P JSR
01188P 04B7 BD 0752 P JSR
01189P 04B9 26 16 040E BNE
01190P 04BD 6F 04 A CLF
01191P 04BF E6 07 A LDY
01192P 04C1 E7 05 A STB
01193P 04C3 27 12 0407 BEQ
01194P 04C5 10A6 02 A LDY
01195P 04C7 8E FC80 A LDA
01196P 04C9 A6 80 A SUB40 LDA
01197P 04CD A7 A0 A STA
01198P 04CF 5A A DECB
01199P 04D1 27 05 0407 BEQ
01200P 04D3 2C F000 A CMPX
01201P 04D5 26 F4 04C8 BNE
01202P 04D7 4F A CLRFA
01203P 04D9 BD 0749 P SUB60 JSR
01204P 04DB BD 073A P JSR
01205P 04DE 39 A SUB70 RTS
01206
01206P 04DE 39 A SUB70 RTS
01207
01207P 04DE 39 A SUB70 RTS
01208
01208P 04DE 39 A SUB70 RTS
01209
01209P 04DE 39 A SUB70 RTS
01210P 04DE 39 A SUB70 RTS
01211P 04DE 39 A SUB70 RTS
01212P 04DE 39 A SUB70 RTS
01213P 04DE 39 A SUB70 RTS
01214P 04DE 39 A SUB70 RTS
01215P 04DE 39 A SUB70 RTS
01216P 04DE 39 A SUB70 RTS
01217P 04DE 39 A SUB70 RTS
01218P 04DE 39 A SUB70 RTS
01219P 04DE 39 A SUB70 RTS
01220
01220P 04DE 39 A SUB70 RTS
01221P 04DE 39 A SUB70 RTS
01222P 04DE 39 A SUB70 RTS
01223P 04DE 39 A SUB70 RTS
01224P 04DE 39 A SUB70 RTS

```

```

01225P 04FF A6 80 A IN225 LDA
01226P 0501 A7 A0 A IN230 STA
01227P 0503 9C A INCB
01228P 0505 33 5F A LEAU
01229P 0507 1183 0000 A CMPO
01230P 0509 27 06 0512 BEQ
01231P 050C 108C F000 A CMPY
01232P 0510 26 ED 04FF BNE
01233P 0512 F7 FC83 A IN240 STB
01234P 0514 4F A CLRFA
01235P 0516 1183 0000 A CMPO
01236P 0518 27 02 051E BEQ
01237P 051C 86 80 A LDA
01238P 051E BA FC81 A IN245 ORA
01239P 0521 B7 FC81 A IN250 STA
01240P 0524 BD 073A P JSR
01241P 0527 BD 071E P JSR
01242P 052A BD 0752 P JSR
01243P 052D 26 30 055F BNE
01244P 052F 1183 0000 A CMPO
01245P 0533 27 0A 053F BEQ
01246P 0535 108E FC80 A LDY
01247P 0539 86 64 A LDA
01248P 053B A7 22 A IN260 STA
01249P 053D 20 ED 04FC BRA
01250
01250P 053D 20 ED 04FC BRA
01251P 053F 35 10 A INP40 PULS
01252P 0541 34 10 A INP50 PSHS
01253P 0543 AE 02 A LDX
01254
01254P 0543 AE 02 A LDX
01255
01255P 0543 AE 02 A LDX
01256P 0545 108E FC80 A LDY
01257P 0549 BD 0752 P JSR
01258P 054C 26 11 055F BNE
01259
01259P 054C 26 11 055F BNE
01260P 054E A6 23 A INP100 LDA
01261P 0550 A7 80 A STA
01262P 0552 CE 0001 A LDY

```

#50 PAPER EMPTY ?  
 #508 LPHC50 YES PE  
 #51 NOT READY,HARD ERROR

\*\*\*\*\*

# SUBSYSTEM OUTPUT  
 #1 SUBSYSTEM INPUT  
 RCBLNH,X RCBLNH MUST BE 0.  
 SUB70  
 RCBLNH,X BYTES NO (1 - 128)  
 SUB70 IF BYTES LENGTH=0,ERROR  
 #128 TOO LONG ?  
 SUB70 YES RCB ERROR  
 X ACCX SAVE RCB ADDR  
 #SHARED DESTINATION  
 RCBDBA,X SOURCE  
 BYHL BUSY CHECK AND HALT SET  
 1,X  
 #50  
 1,X  
 1,Y STATUS READ  
 #57F STATUS RESERVE  
 1,X CONT FLAG SET  
 1,X  
 0,X+ OUT DATA READ  
 0,Y+ DATA STORE  
 SUB20  
 HRESET HALT RESET COMMAND EXEC  
 X ACCX COME BACK (RCB ADDRESS)  
 ASSUME NORMAL RETURN.  
 0,X READ REQUEST NO.  
 #17 SUBIN ?  
 SUB70 NO (SUBOUT)

BYHLWD  
 READST LOAD STATUS.  
 SUB60 ERROR  
 RCBLNH,X INPUT BYTES HIGH=0  
 RCBMT,X NUMBER OF TRANSFER BYTES.  
 RCBLNH,X INPUT BYTES LOWER SET.  
 SUB80 TRANSFER=0 BYTES.  
 RCBDBA,X INPUT DATA TRANSFER DESTINATION  
 #SHARED SOURCE  
 0,X+ READ DATA  
 0,Y+ STORE DATA ON BUFFER

SUB50  
 #SHARED+128  
 SUB40 NORMAL  
 REDYRQ  
 HRESET

\*\*\*\*\*

# CONSOLE LINE INPUT  
 RCBBMH,X INPUT BUFFER LENGTH SET  
 SBT30  
 SINMAX  
 X INDEX SAVE  
 BYHL BUSY CHECK AND HALT SET  
 #SHARED  
 #GET GET COMMAND SET  
 SMCMD,Y C=4 GET COMMAND  
 N=0 SET  
 RCBLNH,X LENGTH GET  
 DATA IS NOTHING ?  
 INZ40  
 RCBDBA,X BUFFER ADDRESS  
 4,Y STRING ADDRESS  
 COUNTER CLEAR (N)

0,X+ X POST INCREMENT  
 0,Y+ BYTE COUNT UP

-1,U  
 #0  
 INZ40  
 #SHARED+128  
 INZ25  
 SHARED+3

#0  
 INZ45  
 #50  
 SHARED+1  
 SHARED+1 S' SET  
 HRESET HALT RESET ,COMMAND OUT  
 BYHLWD BUSY CHECK AND HALT SET  
 READST  
 INP160 ERROR DETECTED  
 #0 END ?  
 INP40 YES END,GO TO INPUT  
 #SHARED CONTINUE  
 #100 C=100 SET  
 SMCMD,Y  
 INZ20

#  
 #SHARED  
 READST STATUS READ.  
 INP160 ERROR  
 3,Y K : RETURN CODE SET  
 0,X+  
 #1 READ DATA COUNT=1 INITIAL

01263P 0558 8D 29 0580 INP150 BSR  
 01264P 0557 26 06 055F BNE  
 01265P 0559 65 FC81 A LDA  
 01266P 055C 28 0C 056A BMI  
 01267P 055E 4F CLRA  
 01268P 055F BD 0749 P INP160 JSR  
 01269P 0562 BD 073A P JSR  
 01270P 0565 35 10 A PULS  
 01271P 0567 EF 04 A STU  
 01272P 0569 39 RTS  
 01273  
 01274P 056A 86 64 A INP200 LDA  
 01275P 056C B7 FC82 A STA  
 01276P 056F BD 073A P JSR  
 01277P 0572 BD 071E P JSR  
 01278P 0575 108E FC80 A LDY  
 01279P 0579 BD 0752 P JSR  
 01280P 057C 26 E1 055F BNE  
 01281P 057E 20 D5 0555 BRA  
 01282  
 01283P 0580 F6 FC84 A P SBTAN EQU  
 01285P 0582 27 13 0598 LDR  
 01286P 0585 108E FC85 A BEQ  
 01287  
 01288P 0589 A6 A0 A A SETR10 LDA  
 01289P 058B 33 41 A LEAU  
 01290P 058D 1183 FFE6 A CMPU  
 01291P 0591 22 07 0594 BHI  
 01292P 0593 A7 80 A STA  
 01293P 0595 5A DECB  
 01294P 0596 26 F1 0589 BNE  
 01295P 0598 4F SBT20 CLRA  
 01296P 0599 39 RTS  
 01297  
 01298P 059A 86 4F A A SBT30 LDA  
 01299P 059C 39 RTS

01301 059D 0580 P INPUT EQU  
 01302P 059D EC 06 A LDD  
 01303P 059F 27 F9 059A BEQ  
 01304P 05A1 FD FFE6 A STD  
 01305P 05A4 BD 0723 P JSR  
 01306P 05A7 108E FC80 A LDY  
 01307P 05A8 86 05 A INZ90 LDA  
 01308P 05AD A7 22 A STA  
 01309P 05AF BD 073A P JSR  
 01310P 05B2 BD 071E P JSR  
 01311P 05B5 20 8A 05A1 BRA  
 01312

01314  
 01315  
 01316 0587 P OUTPUT EQU  
 01317P 0587 4F CLRA  
 01318P 0588 EE 04 A LDU  
 01319P 058A 1183 0000 A CMPU  
 01320P 058E 27 58 0618 BEQ  
 01321P 05C0 BD 0723 P JSR  
 01322P 05C3 108E FC80 A LDY  
 01323P 05C7 86 03 A OUT210 LDA  
 01324P 05C9 A7 22 A STA  
 01325P 05CE AE 02 A LDY  
 01326P 05CD 31 24 A OUT05 LEAY  
 01327P 05CF 5F CLRB  
 01328P 05D0 A6 80 A OUT10 LDA  
 01329P 05D2 A7 A0 A OUT220 STA  
 01330P 05D4 5C INCB  
 01331P 05D5 33 5F A LEAU  
 01332P 05D7 1183 0000 A CMPU  
 01333P 05DE 27 06 05E3 BEQ  
 01334P 05DD 108C FD00 A CMPY  
 01335P 05E1 26 ED 05D0 BNE  
 01336  
 01337P 05E3 F7 FC83 A A OUT20 STB  
 01338P 05E6 86 00 A LDA

01339P 05E8 1183 0000 A CMPU  
 01340P 05EC 27 02 05F0 BEQ  
 01341P 05EE 86 80 A LDA  
 01342P 05F0 8A FC81 A OUT25 ORA  
 01343P 05F3 87 FC81 A OUT230 STA  
 01344P 05F6 BD 073A P JSR  
 01345P 05F9 4F CLRA  
 01346P 05FA 1183 0000 A CMPU  
 01347P 05FE 27 18 0618 BEQ  
 01348P 0600 BD 071E P JSR  
 01349P 0603 BD 0752 P JSR  
 01350P 0606 26 0A 0612 BNE  
 01351P 0608 108E FC80 A LDY  
 01352P 060C 86 64 A LDA  
 01353P 060E A7 22 A OUT240 STA  
 01354P 0610 20 BB 05CD BRA  
 01355  
 01356P 0612 BD 0749 P OUT40 JSR  
 01357P 0615 BD 073A P JSR  
 01358P 0618 39 OUT50 RTS

01360  
 01361  
 01362  
 01363P 0619 BD 0723 P KEYIN EQU  
 01364P 061C 108E FC80 A JSR  
 01365P 0620 86 28 A LDA  
 01366P 0622 A7 22 A KEYZ10 STA  
 01367P 0624 86 00 A LDA  
 01368P 0626 A7 23 A KEYZ20 STA  
 01369P 0628 BD 073A P JSR  
 01370P 062B BD 071E P JSR  
 01371P 062C DE 0002 A LDU  
 01372P 0631 EF 04 A STU  
 01373P 0633 EC 23 A LDD  
 01374P 0635 AE 02 A LDY  
 01375P 0637 ED 84 A STD  
 01376P 0639 BD 0752 P JSR  
 01377P 063C BD 0749 P JSR

SBTRAN SHARED BUFFER TRANSFER  
 INP160 ERROR  
 SHARED+1 CHECK S'  
 INP200 MSB SET CONTINUE  
 REDYRQ  
 HRESET HALT RESET.  
 X ACCX COME BACK  
 RCBLNH,X READ BYTE LENGTH SET  
 RETURN  
 #100  
 SHARED+2 CONT COMMAND SET  
 HRESET HALT RESET  
 BYHLWD BUSY CHECK AND HALT SET  
 #SHARED  
 READST LOAD STATUS.  
 INP160  
 INP150

# SHARED MEMORY TO BUFFER  
 SHARED+4 READ BYTES LENGTH (N)  
 SBT30  
 LDY  
 #SHARED+5 DATA TOP ADDRESS  
 0,Y+ READ DATA  
 1,U INCREMENT U  
 SINMAX BUFFER OVER ERROR ?  
 SBT30 YES  
 0,X+ STORE DATA

SBTR10  
 NORMAL.  
 RETURN.

#73 BUFFER OVER ERROR

# INPUT CONTINUE  
 RCBBMH,X BUFFER MAX LENGTH  
 SBT30  
 SINMAX  
 BYHL BUSY CHECK AND HALT SET  
 #SHARED  
 #GETC GETC COMMAND  
 2,Y  
 HRESET HALT RESET.  
 BYHLWD BUSY CHECK,HALT SET.  
 INP50

# CONSOLE OUTPUT  
 RCBLNH,X LENGTH  
 #0 DATA NOTHING  
 OUT50  
 BYHL BUSY CHECK AND HALT SET  
 #SHARED  
 #PUT  
 SMCMD,Y C=3 PUT COMMAND  
 RCBDBA,X BUFFER ADDRESS  
 4,Y STRING ADDRESS  
 COUNTER CLEAR (N)  
 0,X+  
 0,Y+  
 -1,U  
 #0  
 OUT20  
 BEQ  
 #SHARED+128  
 OUT10  
 SHARED+3 BYTES NUMBER  
 #0

#0  
 OUT25  
 #580  
 SHARED+1  
 SHARED+1 S' SET  
 HRESET HALT RESET  
 NON ERROR  
 END ?  
 OUT50 YES END  
 BYHLWD BUSY CHECK AND HALT SET  
 READST LOAD STATUS.  
 OUT40 ERROR  
 #SHARED  
 #100  
 SMCMD,Y C=100 SET  
 OUT05  
 REDYRQ SUB READY REQUEST.  
 HRESET HALT RESET.

#  
 BYHL BUSY CHECK AND HALT RESET  
 #SHARED  
 #INKEY INKEY COMMAND  
 SMCMD,Y C=41  
 #0  
 SMDAT,Y WAITLESS  
 HRESET HALT RESET  
 BYHLWD BUSY CHECK AND HALT SET  
 #2  
 RCBLNH,X LENGTH SET  
 SMDAT,Y  
 RCBDBA,X  
 0,X  
 STD  
 READST LOAD STATUS.  
 REDYRQ SUB READY REQUEST.

# FM-7 BIOSソース・リスト

01378P	063F	7E	073A	P	JMP	HRESET	HALT RESET,DOUBLE RETURN.
01381P							# CRT HARD COPY
01382P	0642	80	077D	P	HDCOPY	EQU	DUMMY OUTPUT
01383P	0645	C6	01	A	LDE	#1	BLOCK NO=1 INITIAL
01383P	0647	10AE	02	A	LDB	RCE0BA,X	GET BUFFER ADDRESS.
01384							
01385P	064A	34	064A	P	HDLOOP	EQU	#
01386P	064C	E7	A3 00C8	A	STB	BLKNO,Y	BUFFER AND RCE ADDRESS SAVE
01387P	0650	C5	01	A	BITB	#01	BLOCK NO. SAVE
01388P	0652	27	17 066B	E	BEQ	YISET	EVEN ?
01389P	0654	5C					YES EVEN
01390P	0655	86	64	A	YISET	LDA	#100 YUI=100
01391P	0657	A7	A3 00CC	A	STA	PUIY+1,Y	
01392P	0658	5F	A3 00CB	A	CLR	PUIY,Y	UPPER BYTE =0
01393P	065F	86	C7	A	LDA	#199	YLI=199
01394P	0661	A7	A3 00D0	A	STA	PLIY+1,Y	
01395P	0665	6F	A3 00CF	A	CLR	PLIY,Y	UPPER BYTE =0
01396P	0669	20	12 067D	E	BRA	XISET	
01397							
01398P	066B	6F	A3 00CB	A	YISET	CLR	PUIY,Y UPPER BYTE=0
01399P	066F	6F	A3 00CC	A	CLR	PUIY+1,Y	YUI=0
01400P	0673	6F	A3 00CF	A	CLR	PLIY,Y	UPPER BYTE=0
01401P	0677	86	63	A	LDA	#39	
01402P	0679	A7	A3 00D0	A	STA	PLIY+1,Y	YLI=39
01403P	067D	86	04	A	XISET	LDA	#4
01404P	067F	30			MUL		I#4
01405P	0680	83	0008	A	SUED	#8	I#4-8
01406P	0683	ED	A3 00C9	A	STO	PUIX,Y	
01407P	0687	C3	0007	A	ADD	#7	PLIX
01408P	068A	ED	A3 00CD	A	STO	PLIX,Y	
01409							
01410							
01411							
01412P	068E	33	A3 0000	A	DATA INPUT	LEAU	BLACK,Y BLACK,GRAY CONTINUOUS
01413P	0692	86	C8	A	LDA	#200	200 BYTE CLEAR
01414P	0694	ED	061C	P	JSR	CPBUFC	
01415P	0697	A6	04	A	LDA	RCEBDT,X	SOME BLACK COLOR ?
01416P	0699	27	0B 05A6	E	BEQ	GRAYCK	NON BLACK COLOR
01417							
01418P	069B	34	10	A	PSHS	X	RCE ADDRESS SAVE.
01419P	069D	39	A3 0000	A	LEAX	BLACK,Y	
01420P	06A1	ED	07CA	P	JSR	DINPUT	
01421P	06A4	35	10	A	PULS	X	X:COME BACK.
01422P	06A6	A6	0B	A	GRAYCK	LDA	RCEBDT+1,X SOME GRAY COLOR
01423P	06A8	27	07 06B1	E	BEQ	DOUT	NON GRAY COLOR
01424							
01425P	06AA	30	A3 0064	A	GRAY INPUT	LEAX	GRAY,Y
01426P	06AE	ED	07CA	P	JSR	DINPUT	
01427							
01428							
01429P	06B1	A6	A3 00C8	A	DATA OUTPUT	LDA	TO PRINTER
01430P	06B5	85	01	A	DOUT	BLKNO,Y	
01431P	06B7	27	03 06BC	A	BITA	#01	
01432P	06B9	ED	0758	P	BEQ	DATA0T	
01433P	06BC	30	0758	P	JSR	LFINCH	ODD.LF PITCH SET,AND BLANK OUT
01434P	06C0	31	A3 0084	A	DATA0T	LEAX	BLACK+100,Y BLACK=IX
01435P	06C4	86	1E	A	LEAY	GRAY+100,Y GRAY=IY	
01436P	06C6	ED	07A3	P	JSR	#ESC	BI-PRINT MODE SET
01437P	06C9	86	4C	A	LPPUT	#L	
01438P	06CB	ED	07A3	P	JSR	LPPUT	
01439P	06CE	86	30	A	LDA	#00	LOWER
01440P	06D0	ED	07A3	P	JSR	LPPUT	\$0190=400 DOTS
01441P	06D3	86	01	A	LDA	#01	UPPER
01442P	06D5	ED	07A3	P	JSR	LPPUT	
01443P	06D8	A6	82	A	DATA01	0,-X	BLACK DATA READ
01444P	06DA	ED	07A3	P	JSR	LPPUT	
01445P	06DD	A6	84	A	LDA	0,-X	
01446P	06DF	AA	A2	A	ORA	0,-Y	
01447P	06E1	ED	07A3	P	JSR	LPPUT	BLACK,GRAY
01448P	06E4	A6	84	A	LDA	0,-X	AGAIN SAME DATA
01449P	06E6	ED	07A3	P	JSR	LPPUT	
01450P	06E9	A6	84	A	LDA	0,-X	POINTER INCREMENT (BLACK)
01451P	06EB	AA	A4	A	ORA	0,-Y	POINTER INCREMENT (GRAY)
01452P	06ED	ED	07A3	P	JSR	LPPUT	
01453P	06F0	AC	62	A	CMPLX	2,S	100 BYTES END ?
01454P	06F2	28	E4 06D8	E	BNE	DATA01	NO,CONTINUE
01455P	06F4	35	30	A	PULS	Y,X	Y,X COME BACK
01456P	06F6	E8	A3 00C8	A	LDB	BLKNO,Y	EVEN BLOCK ?
01457P	06FA	C5	01	A	BITB	#01	
01458P	06FC	26	02 0700	E	BNE	#+4	NO,ODD
01459P	06FE	8D	6E 076E	E	BSR	PCRLF	LINEFEED OUTPUT
01460							
01461							
01462P	0700	5C					
01463P	0701	C1	A1	A	CMPLX	#161	ALL BLOCK END ?
01464P	0703	1026	FF43 064A	E	BNE	HDLOOP	NO,CONTINUE
01465P	0707	8D	07 0710	E	LDB	LFP1.6	1/6 INCH LF PITCH SET.
01466P	0709	88	0C	A	LDA	#FF	FORM FEED
01467P	070B	ED	07A3	P	JSR	LPPUT	
01468P	070E	4F			CLRA		
01469P	070F	39			DATA05 RTS		
01470							
01471							
01472P	0710	86	0710	P	LFP1.6	EQU	#
01473P	0712	8D	18	A	LDA	#ESC	ESC+A+12
01474P	0714	27	75 0789	E	BEQ	LPPUT	\$1E
01475P	0716	88	F9 070F	E	BSR	DATA05	
01476P	0718	86	41	A	LDA	#1A	A
01477P	071A	86	0C 0789	E	BSR	LPPUT	
01478P	071C	20	6E 0789	E	BRA	#12	12 DOTS PITCH SET.
01479							
01480							
01481P	071E	86	071E	P	BYHLWD	EQU	#
01482P	0720	4A	18	A	DLYLOP	LDA	#24 TIME DELAY.
01483P	0721	26	FD 0720	E	BNE	DECA	ABOUT 50 MICRO SEC.
01484							2 CYCLE.
01485P	0723	36	0723	P	BYHL	EQU	3 CYCLE.
01486P	0725	86	05	A	LDA	#BSIN	READY CHECK AND HALT SET
01487P	0727	1A	FC 0723	E	BNE	BYHL	?
01488P	0729	86	80	A	ORCC	#040	NO CONTINUE
01489P	072B	97	05	A	HLSET	LDA	FIRQ IS MASKED UNTIL HALT RESET.++
01490P	072D	34	04	A	STA	SESOUT	
01491P	072F	86	04	A	PSHS	R.	

01491P	072F	C6		08	A			
01492P	0731	5A				HLTOVR	LDB	
01493P	0732	27		04	0738	*	REQ	
01494P	0733	26		06	A		RQA	
01495P	0736	2A	F9	0731			EPL	
01496P	0738	35		84	A	HLTRTS	PULS	
01497					*			
01498								
01499				073A	P	HRESET	EQU	
01500P	073A	34		02	A		PSHS	
01501P	073C	36		00	A		LDA	
01502P	073E	37		05	A		STA	
01503P	0740	1C	BF	A			ANDCU	
01504P	0742	35		02	A		PULS	
01505P	0744	3D		02	0748 HRE10		BSR	
01506P	0746	8D		00	0748		ESR	
01507P	0748	39					RTS	
01508						#		
01509				0749	P	* REDYQO	EQU	
01510P	074C	FC	FC80		A		LDR	
01511P	074B	CA	80		A		ORB	
01512P	074E	F7	FC80		A		STB	
01513P	0751	39					RTS	
01514					*			
01515				0752	P	* READST	EQU	
01516P	0752	B6	FC80		A		LDA	
01517P	0755	34	7F		A		ANDU	
01518P	0757	39					RTS	
01519					*			
01520				0758	P	* LF INCH	EQU	
01521P	0758	36	1B		A		LDA	
01522P	075A	8D	40	07A3			ESR	
01523P	075C	36	41		A		LDA	
01524P	075E	8D	49	07A3			ESR	
01525P	0760	36	08		A		LDA	
01526P	0762	8D	45	07A3			BSR	
01527P	0764	C6	07		A		LDB	
01528P	0766	36	20		A	L FFINIO	LDA	
01529P	0768	8D	3F	07A3			ESR	
01530P	076A	5A					DECB	
01531P	076E	26	F9	0766			BNE	
01532P	076D	39				LFRITS	RTS	
01533					*			
01534								
01535								
01536P	076E	36	0A		P	* PCRLF	EQU	
01537					*			
01538				00FF	A		I FEQ	
01539						ENDC		
01540					#			
01541								
01542				0000	A		I FEQ	
01543P	0770	8D	40	07B2			ESR	
01544P	0772	7D	0313	A			TST	
01545P	0775	27	F6	07BD			BEQ	
01546					P	AETRN	JMP	
01547P	0777	8D	0777		P		JSR	
01548P	077A	7E	0000		A		JMP	
01549							ENDC	
01550				077D	P	* PCRLFX	EQU	
01551P	077D	86	0A		A		LDA	
01552				00FF	A		I FEQ	
01553						ENDC		
01554					*			
01555				0000	A		I FEQ	
01556P	077F	8D	31	07B2			LOX	
01558P	0781	7D	0313	A			TST	
01559P	0784	27	E7	07BD			BEQ	
01560P	0786	7E	0000	A			JMP	
01561							ENDC	
01562					*			
01563				0789	P	* LPPUTT	EQU	
01564P	0789	34	04		A		PSHS	
01565P	078B	0E	02		A		LDB	
01566P	078D	C5			A		BITB	
01567P	078F	27	37	07C8			BEQ	
01568P	0791	3E	FFFF	A			LOX	
01569P	0794	34	10		A	LPL0OP	PSHS	
01570P	0796	8E	0012	A			LDB	
01571P	079B	6D	01B2	P			LOX	
01572P	079C	35	10		A		PULS	
01573P	079E	30	1F	A			LEAX	
01574P	07A0	27	26	07C8			BEQ	
01575P	07A2	06	02	A			LDB	
01576P	07A4	25					LSRBE	
01577P	07A5	25	ED	0794			BCE	
01578P	07A7	20	15	07BE			ERA	
01579					*			
01580P	07A9			07A9	P	* LPPUT	EQU	
01581P	07A3	34	04		A		PSHS	
01582P	07AB	06	02		A	LPPUTO	LSRBE	
01583P	07AD	94						
01584P	07AE	25	F8	07AB			BCE	
01585P	07B0	20	0C	07BE			BFA	
01586					*			
01587				07B2	P	* LPPUTX	EQU	
01588P	07B2	34	04		A		PSHS	
01589				07B4	P	* LPPUT1	EQU	
01591					*			
01592								
01593P	07B4	7D	0313	A			I FEQ	
01594P	07B7	26	0F	07C8			TST	
01595							ENE	
01596					*		ENDC	
01598P	07B9	D6	02		A		LDB	
01599P	07BB	54					LSRBE	
01600P	07BC	25	F6	07B4			BCE	
01601				07BE	P	* LPPUT2	EQU	
01602P	07BE	37	01	A			STA	
01603P	07C0	C5	00		A		STB	
01604P	07C2	00			A		STB	
01605P	07C4	C6	40		A		LDB	
01606P	07C6	07	00		A		STB	



# FM-7 BIOSソース・リスト

```

01718P 088E A7 02 A STA SMCMD,X
01719P 08A0 EC A3 00C9 A LDD PUIX,Y PUIX STORE
01720P 08A4 ED 03 A STD 3,X
01721P 08A6 EC A3 00CB A LDD PUIY,Y PUIY STORE
01722P 08AA ED 05 A STD 5,X
01723P 08AC EC A3 00CD A LDD PLIX,Y PLIX STORE
01724P 08B0 ED 07 A STD 7,X
01725P 08B2 EC A3 00CF A LDD PLIY,Y PLIY STORE
01726P 08B6 ED 09 A STD 9,X
01727P 08B8 ED 09 A STD 9,X
01728P 08B8 B6 FFE6 A LDA SINMAX ++ GET SCREEN STATUS.
01729P 08BE 34 02 A PSHS A ++ PUSH IT.
01730P 08D0 86 0C A LDA #12 ++ SET GET BLOCK1 COLOR CODE TABLE ADDRESS.
01731P 08E6 5F 05 A CLRE ++ SET COLOR CODE.
01732P 08C0 64 E4 A SDIN11 LSR ++ SHIFT 1 BIT.
01733P 08C2 24 03 08C7 P BCC SDIN12 ++ IS IT 0 ? YES,SKIP IT.
01734P 08C4 E7 85 P STB A,X ++ STORE COLOR CODE.
01735P 08C6 C4 05 A INCA ++ COLOR NUMBER +1
01736P 08C7 5C 05 A SDIN12 INCB ++ NEXT COLOR CODE.
01737P 08C8 C1 08 A CMPE #8 ++ ALL COLOR END ?
01738P 08CA 26 F4 08C0 A ENE SDIN11 ++ NO,NEXT COLOR.
01739P 08CC 80 0C A SUEA #12 ++ CALC SET UP COLOR NUMBER.
01740P 08CE A7 0E A STA 11,X ++ STORE SET UP COLOR NUMBER.
01741P 08D0 32 81 A LEAS 1,S ++ ADJUST STACK.
01742P 08D2 BD 073A P JSR HRESET HALT RESET
01743P 08D5 BD 071E P JSR BYHLWD IF COMMAND END,HALT SET.
01744P 08D8 30 04 A LEAX 4,X
01745P 08DA 31 A3 0000 A LEAY BLACK,Y
01746P 08DE A6 80 A SDIN30 LDA 0,X+ DATA READ.
01747P 08E0 A7 A0 A STA 0,Y+ STORE DATA
01748P 08E2 8C FCD4 A CMPX #SHARED+84 80 BYTE END ?
01749P 08E5 26 F7 08DE A SDIN30 BNE
01750P 08E7 ED 0749 P JSR REDYRQ SUB READY REQUEST.
01751P 08EA BD 073A P JSR HRESET HALT RESET.
01752P 08ED 35 A0 A PULS Y,PC
01753P 08EF 34 08EF P MODIFY EQU # DATA MODIFICATION
01754P 08F4 33 A4 A LCAU Y Y->U
01755P 08F6 30 A3 0000 A LEAX BLACK,Y DATA BUFFER ADDRESS.
01756P 08F8 86 0A A 0064 A LEAY GRAY,Y RESULT OF MODIFICATION.
01757P 08FD 34 02 A DTL00P PSHS #10 COUNT=10
01758P 08FF 5F 85 A DTL0PX CLRB A COUNTER SAVE.
01759P 0900 A6 20 0924 A BSR B,X
01760P 0902 80 0A A ADDB #10 SHIFT
01761P 0904 C8 50 A CMPE #80
01762P 0906 26 F6 0900 A BNE DTL0PX
01763P 0908 34 10 A X PSHS X
01764P 090A C6 08 A LDB #8 X SAVE
01765P 090C 30 C9 005A A LEAX BLACK+90,U RESULT BUFFER.
01766P 090E A6 80 A DTL010 LDA 0,X+
01767P 0910 A7 A0 A STA 0,Y+
01768P 0912 5A 08 A DECB
01769P 0914 26 F9 0912 A BNE DTL010
01770P 0916 5A 10 A PULS X X COME BACK
01771P 0918 35 01 A LEAX 1,X X INCREMENT
01772P 091A 30 02 A PULS A COUNTER COME BACK.
01773P 091C 4A 0E A DECA
01774P 091E 26 B9 09FD A BNE DTL00P
01775P 0920 35 A0 A PULS Y,PC CONTINUE RETURN
01776P 0922 35 A0 A PULS Y,PC CONTINUE RETURN
01777P 0924 48 0924 P SHIFT EQU #
01778P 0926 48 0924 P SHIFT EQU #
01779P 0928 48 0924 P SHIFT EQU #
01780P 0930 48 0924 P SHIFT EQU #
01781P 0932 48 0924 P SHIFT EQU #
01782P 0934 48 0924 P SHIFT EQU #
01783P 0936 48 0924 P SHIFT EQU #
01784P 0938 48 0924 P SHIFT EQU #
01785P 0940 48 0924 P SHIFT EQU #
01786P 0942 48 0924 P SHIFT EQU #
01787P 0944 48 0924 P SHIFT EQU #
01788P 0946 48 0924 P SHIFT EQU #
01789P 0948 48 0924 P SHIFT EQU #
01790P 0950 48 0924 P SHIFT EQU #
01791P 0952 48 0924 P SHIFT EQU #
01792P 0954 48 0924 P SHIFT EQU #
01793P 0956 48 0924 P SHIFT EQU #
01794P 0958 48 0924 P SHIFT EQU #
01795P 0960 48 0924 P SHIFT EQU #
01796P 0962 48 0924 P SHIFT EQU #
01797P 0964 48 0924 P SHIFT EQU #
01798P 0966 48 0924 P SHIFT EQU #
01799P 0968 48 0924 P SHIFT EQU #
01800P 0970 48 0924 P SHIFT EQU #
01801P 0972 48 0924 P SHIFT EQU #
01802P 0974 48 0924 P SHIFT EQU #
01803P 0976 48 0924 P SHIFT EQU #
01804P 0978 48 0924 P SHIFT EQU #
01805P 0980 48 0924 P SHIFT EQU #
01806P 0982 48 0924 P SHIFT EQU #
01807P 0984 48 0924 P SHIFT EQU #
01808P 0986 48 0924 P SHIFT EQU #
01809P 0988 48 0924 P SHIFT EQU #
01810P 0990 48 0924 P SHIFT EQU #
01811P 0992 48 0924 P SHIFT EQU #
01812P 0994 48 0924 P SHIFT EQU #
01813P 0996 48 0924 P SHIFT EQU #
01814P 0998 48 0924 P SHIFT EQU #
01815P 1000 48 0924 P SHIFT EQU #
01816P 1002 48 0924 P SHIFT EQU #
01817P 1004 48 0924 P SHIFT EQU #
01818P 1006 48 0924 P SHIFT EQU #
01819P 1008 48 0924 P SHIFT EQU #
01820P 1010 48 0924 P SHIFT EQU #
01821P 1012 48 0924 P SHIFT EQU #
01822P 1014 48 0924 P SHIFT EQU #
01823P 1016 48 0924 P SHIFT EQU #
01824P 1018 48 0924 P SHIFT EQU #
01825P 1020 48 0924 P SHIFT EQU #
01826P 1022 48 0924 P SHIFT EQU #
01827P 1024 48 0924 P SHIFT EQU #
01828P 1026 48 0924 P SHIFT EQU #
01829P 1028 48 0924 P SHIFT EQU #
01830P 1030 48 0924 P SHIFT EQU #
01831P 1032 48 0924 P SHIFT EQU #
01832P 1034 48 0924 P SHIFT EQU #
01833P 1036 48 0924 P SHIFT EQU #
01834P 1038 48 0924 P SHIFT EQU #
01835P 1040 48 0924 P SHIFT EQU #
01836P 1042 48 0924 P SHIFT EQU #
01837P 1044 48 0924 P SHIFT EQU #
01838P 1046 48 0924 P SHIFT EQU #
01839P 1048 48 0924 P SHIFT EQU #
01840P 1050 48 0924 P SHIFT EQU #
01841P 1052 48 0924 P SHIFT EQU #
01842P 1054 48 0924 P SHIFT EQU #
01843P 1056 48 0924 P SHIFT EQU #
01844P 1058 48 0924 P SHIFT EQU #
01845P 1060 48 0924 P SHIFT EQU #
01846P 1062 48 0924 P SHIFT EQU #
01847P 1064 48 0924 P SHIFT EQU #
01848P 1066 48 0924 P SHIFT EQU #
01849P 1068 48 0924 P SHIFT EQU #
01850P 1070 48 0924 P SHIFT EQU #
01851P 1072 48 0924 P SHIFT EQU #
01852P 1074 48 0924 P SHIFT EQU #
01853P 1076 48 0924 P SHIFT EQU #
01854P 1078 48 0924 P SHIFT EQU #
01855P 1080 48 0924 P SHIFT EQU #
01856P 1082 48 0924 P SHIFT EQU #
01857P 1084 48 0924 P SHIFT EQU #
01858P 1086 48 0924 P SHIFT EQU #
01859P 1088 48 0924 P SHIFT EQU #
01860P 1090 48 0924 P SHIFT EQU #
01861P 1092 48 0924 P SHIFT EQU #
01862P 1094 48 0924 P SHIFT EQU #
01863P 1096 48 0924 P SHIFT EQU #
01864P 1098 48 0924 P SHIFT EQU #
01865P 1100 48 0924 P SHIFT EQU #
01866P 1102 48 0924 P SHIFT EQU #
01867P 1104 48 0924 P SHIFT EQU #
01868P 1106 48 0924 P SHIFT EQU #
01869P 1108 48 0924 P SHIFT EQU #
01870P 1110 48 0924 P SHIFT EQU #
01871P 1112 48 0924 P SHIFT EQU #
01872P 1114 48 0924 P SHIFT EQU #
01873P 1116 48 0924 P SHIFT EQU #
01874P 1118 48 0924 P SHIFT EQU #
01875P 1120 48 0924 P SHIFT EQU #
01876P 1122 48 0924 P SHIFT EQU #
01877P 1124 48 0924 P SHIFT EQU #
01878P 1126 48 0924 P SHIFT EQU #
01879P 1128 48 0924 P SHIFT EQU #
01880P 1130 48 0924 P SHIFT EQU #
01881P 1132 48 0924 P SHIFT EQU #
01882P 1134 48 0924 P SHIFT EQU #
01883P 1136 48 0924 P SHIFT EQU #
01884P 1138 48 0924 P SHIFT EQU #
01885P 1140 48 0924 P SHIFT EQU #
01886P 1142 48 0924 P SHIFT EQU #
01887P 1144 48 0924 P SHIFT EQU #
01888P 1146 48 0924 P SHIFT EQU #
01889P 1148 48 0924 P SHIFT EQU #
01890P 1150 48 0924 P SHIFT EQU #
01891P 1152 48 0924 P SHIFT EQU #
01892P 1154 48 0924 P SHIFT EQU #
01893P 1156 48 0924 P SHIFT EQU #
01894P 1158 48 0924 P SHIFT EQU #
01895P 1160 48 0924 P SHIFT EQU #
01896P 1162 48 0924 P SHIFT EQU #
01897P 1164 48 0924 P SHIFT EQU #
01898P 1166 48 0924 P SHIFT EQU #
01899P 1168 48 0924 P SHIFT EQU #
01900P 1170 48 0924 P SHIFT EQU #
01901P 1172 48 0924 P SHIFT EQU #
01902P 1174 48 0924 P SHIFT EQU #
01903P 1176 48 0924 P SHIFT EQU #
01904P 1178 48 0924 P SHIFT EQU #
01905P 1180 48 0924 P SHIFT EQU #
01906P 1182 48 0924 P SHIFT EQU #
01907P 1184 48 0924 P SHIFT EQU #
01908P 1186 48 0924 P SHIFT EQU #
01909P 1188 48 0924 P SHIFT EQU #
01910P 1190 48 0924 P SHIFT EQU #
01911P 1192 48 0924 P SHIFT EQU #
01912P 1194 48 0924 P SHIFT EQU #
01913P 1196 48 0924 P SHIFT EQU #
01914P 1198 48 0924 P SHIFT EQU #
01915P 1200 48 0924 P SHIFT EQU #
01916P 1202 48 0924 P SHIFT EQU #
01917P 1204 48 0924 P SHIFT EQU #
01918P 1206 48 0924 P SHIFT EQU #
01919P 1208 48 0924 P SHIFT EQU #
01920P 1210 48 0924 P SHIFT EQU #
01921P 1212 48 0924 P SHIFT EQU #
01922P 1214 48 0924 P SHIFT EQU #
01923P 1216 48 0924 P SHIFT EQU #
01924P 1218 48 0924 P SHIFT EQU #
01925P 1220 48 0924 P SHIFT EQU #
01926P 1222 48 0924 P SHIFT EQU #
01927P 1224 48 0924 P SHIFT EQU #
01928P 1226 48 0924 P SHIFT EQU #
01929P 1228 48 0924 P SHIFT EQU #
01930P 1230 48 0924 P SHIFT EQU #
01931P 1232 48 0924 P SHIFT EQU #
01932P 1234 48 0924 P SHIFT EQU #
01933P 1236 48 0924 P SHIFT EQU #
01934P 1238 48 0924 P SHIFT EQU #
01935P 1240 48 0924 P SHIFT EQU #
01936P 1242 48 0924 P SHIFT EQU #
01937P 1244 48 0924 P SHIFT EQU #
01938P 1246 48 0924 P SHIFT EQU #
01939P 1248 48 0924 P SHIFT EQU #
01940P 1250 48 0924 P SHIFT EQU #

```

```

01829P 0958 BD 07A3 P JSR
01830P 0960 86 50 A LDA
01831P 0962 BD 07A3 P JSR
01832P 0964 86 00 A LDA
01833P 0966 BD 07A3 P JSR
01834P 0968 30 A3 0064 A BITOUT LEAX
01835P 0970 C6 50 A LDA
01836P 0972 86 80 A NEXLPD LDA
01837P 0974 BD 07A3 P JSR
01838P 0976 5A 01 A DECB
01839P 0978 26 F8 097B BNE
01840P 0980 C5 A3 00C8 A LDA
01841P 0982 26 08 A CMPA
01842P 0984 81 03 098E BNE
01843P 0986 26 076E P JSR
01844P 0988 BD 076E P JSR
01845P 0990 39 098F P NOCRLF RTS
01846P 0992 36 01 A BIINIT LDA
01847P 0994 97 03 A STA
01848P 0996 0F 01 A CLR
01849P 0998 86 40 A LDA
01850P 0999 97 00 A STA
01851P 0999 4F 00 A CLRA
01852P 0999 4F 00 A CLRA
01853P 0999 4F 00 A CLRA
01854P 0999 4F 00 A CLRA
01855P 0999 4F 00 A CLRA
01856P 0999 4F 00 A CLRA
01857P 0999 4F 00 A CLRA
01858P 0999 4F 00 A CLRA
01859P 0999 4F 00 A CLRA
01860P 0999 4F 00 A CLRA
01861P 0999 4F 00 A CLRA
01862P 0999 4F 00 A CLRA
01863P 0999 4F 00 A CLRA
01864P 0999 4F 00 A CLRA
01865P 0999 4F 00 A CLRA
01866P 0999 4F 00 A CLRA
01867P 0999 4F 00 A CLRA
01868P 0999 4F 00 A CLRA
01869P 0999 4F 00 A CLRA
01870P 0999 4F 00 A CLRA
01871P 0999 4F 00 A CLRA
01872P 0999 4F 00 A CLRA
01873P 0999 4F 00 A CLRA
01874P 0999 4F 00 A CLRA
01875P 0999 4F 00 A CLRA
01876P 0999 4F 00 A CLRA
01877P 0999 4F 00 A CLRA
01878P 0999 4F 00 A CLRA
01879P 0999 4F 00 A CLRA
01880P 0999 4F 00 A CLRA
01881P 0999 4F 00 A CLRA
01882P 0999 4F 00 A CLRA
01883P 0999 4F 00 A CLRA
01884P 0999 4F 00 A CLRA
01885P 0999 4F 00 A CLRA
01886P 0999 4F 00 A CLRA
01887P 0999 4F 00 A CLRA
01888P 0999 4F 00 A CLRA
01889P 0999 4F 00 A CLRA
01890P 0999 4F 00 A CLRA
01891P 0999 4F 00 A CLRA
01892P 0999 4F 00 A CLRA
01893P 0999 4F 00 A CLRA
01894P 0999 4F 00 A CLRA
01895P 0999 4F 00 A CLRA
01896P 0999 4F 00 A CLRA
01897P 0999 4F 00 A CLRA
01898P 0999 4F 00 A CLRA
01899P 0999 4F 00 A CLRA
01900P 0999 4F 00 A CLRA
01901P 0999 4F 00 A CLRA
01902P 0999 4F 00 A CLRA
01903P 0999 4F 00 A CLRA
01904P 0999 4F 00 A CLRA
01905P 0999 4F 00 A CLRA
01906P 0999 4F 00 A CLRA
01907P 0999 4F 00 A CLRA
01908P 0999 4F 00 A CLRA
01909P 0999 4F 00 A CLRA
01910P 0999 4F 00 A CLRA
01911P 0999 4F 00 A CLRA
01912P 0999 4F 00 A CLRA
01913P 0999 4F 00 A CLRA
01914P 0999 4F 00 A CLRA
01915P 0999 4F 00 A CLRA
01916P 0999 4F 00 A CLRA
01917P 0999 4F 00 A CLRA
01918P 0999 4F 00 A CLRA
01919P 0999 4F 00 A CLRA
01920P 0999 4F 00 A CLRA
01921P 0999 4F 00 A CLRA
01922P 0999 4F 00 A CLRA
01923P 0999 4F 00 A CLRA
01924P 0999 4F 00 A CLRA
01925P 0999 4F 00 A CLRA
01926P 0999 4F 00 A CLRA
01927P 0999 4F 00 A CLRA
01928P 0999 4F 00 A CLRA
01929P 0999 4F 00 A CLRA
01930P 0999 4F 00 A CLRA
01931P 0999 4F 00 A CLRA
01932P 0999 4F 00 A CLRA
01933P 0999 4F 00 A CLRA
01934P 0999 4F 00 A CLRA
01935P 0999 4F 00 A CLRA
01936P 0999 4F 00 A CLRA
01937P 0999 4F 00 A CLRA
01938P 0999 4F 00 A CLRA
01939P 0999 4F 00 A CLRA
01940P 0999 4F 00 A CLRA

```

LPPUT LOWER=80 DOTS.  
#50 LPPUT UPPER=0  
#0 LPPUT

GRAY,Y LP PUT DATA TOP ADDRESS.  
#80 COUNTER SET.  
0,X+ GET A DATA  
LPPUT PRINT OUT  
ALL DATA END?  
NO,NEXT LP DATA.  
BLINO,Y LAST BLOCK OF LINE?  
#8 NOCLRF  
PCRLF CR LF  
PCRLF RETURN  
# ETOS INITIAL SETTING  
#01 SPEAKER SWITCH ON  
BELL

LDPOT DUMMY DATA=00  
#540 STROBE="H"  
MLPOT DUMMY OUTPUT  
NORMAL END.

COPY WORK  
#00 BLACK DATA BUFFER OFFSET.  
#54 GRAY DATA BUFFER OFFSET.  
#C8 CRT DATA BLOCK NO.

#C9 PUIX ADDRESS  
#C8 PUIY ADDRESS

#CD PLIX ADDRESS  
#CF PLIY ADDRESS

#FEFC BASIC RESTART ADDRESS. DON'T INIT.  
#FEFE BASIC START ADDRESS. FROM INITIALIZE.  
BOOT ASSEMBLE WHEN BOOT=0  
L

FBASIC MODE(NO 0) BOOTSTRAP LOADER

#FE00  
# FBASIC BOOT PROGRAM.  
STARTS

#FC7F  
#0100  
#FRESTR  
#DWRIT  
#DREAD  
#02 STOP  
#02 STOP KEY PRESSED?  
DISK DISK CODE LOAD TRY.  
ESRES YES,JUMP TO WARM START ENTRY.  
SYSJMP

# TRY TO READ DISK CODE.  
#BSOUT

NONDS2  
#5 RETRY COUNTS=5 SET.  
# #BOTRCB  
#DREAD RCB ADDRESS FOR BOOT.  
#DREAD FLOPPY READ ROUTINE.  
# GET RETRY COUNTS  
#IPLADR

SYSJMP ANY ERROR?  
#0A DISK IPL LOAD COMPLETED.  
NONDS2 NOT READY?

LRETZZ  
#0MENT

DWRITE MOTOR STOP.  
ROM BASIC INTERFACE=ACCA=0  
A,DP DP=0  
0,X GO TO EACH ENTRY.

#  
#RETRCB  
#RSTOR  
#  
#LRETRY

#  
#DREAD RQNO  
#0 STATUS  
#IPLADR TRACK=0  
#1 SECTOR=1  
#0 SIDE=0  
#0 UNIT=0

#  
#DREAD RQNO  
#0,0,0

# UNIT 0 SET.

BOOT=1 ASSEMBLE WHEN BOOT=1 : BUBBLE MODE.

01842 FFE5 A IFEQ BOOT=2 ASSEMBLE WHEN BOOT=2.  
01843 ENDC  
01845 TTL 5 INCH FD DRIVER MODULE (FEASIC : ROM-BASED  
MODE)

01846  
01847  
01848  
01849  
01850  
01851 FE51 A RRESTP EQU # SEEK TRACK 0.  
01852 FE61 CE FFE1 A LDU #CURTRK,  
01853 FE64 A5 07 A LLA RCBUNT,X  
01854 FE66 A PRESEN EQU #  
01855 FE66 84 03 A ANDA #C8 UNIT SELECT.  
01856 FE68 8F C6 A CLR A,U TRACK ADDRESS 0 SET.  
01857 FE6A D5 1D A LDR DDRIVE  
01858 FE6C 34 80 A ORA #50  
01859 FE6E 97 1D A STA DDRIVE  
01860 FE70 3D 48 FE6A A BSR CME15  
01861 FE72 36 1D A LLA DDRIVE  
01862 FE74 2A 48 FE61 A BPL CME15  
01863 FE76 5D TSTB  
01864 FE77 28 08 FE61 BMI Z1000  
01865 FE79 85 02 A LLA #2  
01866 FE7B 8D FF30 A JSR TMS1  
01867 FE7D 4A DECA  
01868 FE7F 26 FA FE7B SNE Z500  
01869 FE81 86 0A FE7B LDA #30A RESTORE COMMAND (V=0)  
01870 FE83 20 1A FE9F BRA CMEXEC BRANCH COMMAND EXECUTION

01871  
01872 FE35 A SEEK EQU # SEEK TRACK  
01873 FE38 89 FE70 BSR ORSEL DRIVE SELECT  
01874 FE37 25 38 FE61 BCS CME16  
01875 FE39 CE FFE1 A LDU #CURTRK  
01876 FE3C F5 FFE8 A LDR DSTDRV  
01877 FE3F A5 C5 A LLA B,U  
01878 FE31 97 19 A STA DTRACK

01879 FE33 A5 04 A LLA RCBETR,X DESTINATION TRACK LOAD.  
01880 FE35 A1 C5 A CMPA B,U  
01881 FE37 27 20 FE68 BEQ CME11 NON SEEK ACTION  
01882 FE39 A7 C5 A STA B,U NEW TRACK ADDRESS SET.  
01883 FE3B 37 1B A STA TRACK NO IN DATA REG  
01884 FE3D 36 1E A LLA #1E SEEK COMMAND SET (V=1)  
01885 FE3F A CMEXEC EQU # COMMAND EXECUTION  
01886 FE3F 18 A STA DCMD COMMAND STORE  
01887 FE41 86 06 A LLA #6 3 --> 6  
01888 FE43 4A 1F FE68 BEQ CME25 TIME OVER ?  
01889 FE45 27 1F FE68 BEQ CME05 YES, TIME OVER.  
01890 FE47 108E C800 A CME05 LDY #C800 ABOUT 0.5 SECOND  
01891 FE4A 31 3F A CME10 LEAY -1,Y  
01892 FE4C 27 F5 FE63 BEQ CME02  
01893 FE4E D6 1F A LDR DDRO READ DRQ,IRQ  
01894 FE50 C5 40 A BITB #540 IRQREQ ?  
01895 FE52 27 F6 FE6A BEQ CME10 BUSY, LOOP  
01896 FE54 D6 18 A LDR DSTAT MAKE SURE  
01897 FE56 6D FF53 A JSR TM20M  
01898 FE58 4F CME11 CLRA

01899 FE5A 17 0000 FE6D CME15 LBSR #+3  
01900 FE5B 17 0000 FE6D LBSR #+3  
02001A FE60 33 0A A CME15 LLA #0A NOT READY  
02002A FE63 20 02 FE67 BRA CME30  
02003A FE65 86 0F A CME25 LLA #0F TIME OVER ERROR  
02004A FE67 1A 01 A CME30 SEC  
02005A FE69 39 RTS  
02006A FE6B 18 A FE6A EQU # FREADY  
02007A FE6C D6 18 A LDR DSTAT  
02008A FE6E 24 04 FE63 BCC YESRDY  
02009A FE6F 24 04 FE63 BCC YESRDY  
02010A FE70 24 04 FE63 BCC YESRDY  
02011A FE71 24 04 FE63 BCC YESRDY  
02012A FE72 24 04 FE63 BCC YESRDY  
02013A FE73 24 04 FE63 BCC YESRDY

02014A FE7C C6 D0 A LDR STB #500 FORCE INTERRUPT COMMAND.  
02015A FE7D D7 18 A STB DCMD  
02016 FE7E 18 A # FDC READY  
02017A FE7F 36 18 A YESRDY LLA DDATA  
02018A FE80 43 60 FE6A COMA  
02019A FE81 37 18 A STA DDATA  
02020A FE82 80 60 FE6A BSR CME15  
02021A FE83 97 18 A LLA DDATA  
02022A FE84 26 06 FE63 SNE CME11  
02023A FE85 CE A000 A LDU #A000  
02024A FE86 33 5F A RDCYCHK LEAU -1,U  
02025A FE87 1183 0000 A CMU #0  
02026A FE88 27 06 FE6F BEQ TIMOUT YES TIMEOUT Z=1 ZERO.  
02027A FE89 D6 18 A LDR DSTAT  
02028A FE8B 28 F4 FE61 BMI RDCYCHK  
02029A FE8D C6 01 A LDR #1  
02030A FE8F 39 TIMOUT RTS  
02031A FE90 39 TIMOUT RTS  
02032A FE91 39 TIMOUT RTS  
02033A FE92 39 TIMOUT RTS  
02034A FE93 39 TIMOUT RTS  
02035A FE94 D6 FE6F A ORSEL EQU #  
02036A FE95 F7 FFE5 A LDR DDRIVE  
02037A FE96 F5 C4 03 A ANDB #03  
02038A FE97 F7 FFE0 A STB CRDRIV  
02039A FE98 F5 C4 FFE1 A LDU #CURTRK  
02040A FE99 A6 07 A LLA RCBUNT,X  
02041A FE9A 84 03 A ANDA #03  
02042A FE9B 87 FFE8 A STA DSTDRV  
02043A FE9C 7D FFE8 A DMTOR WAS ON ?  
02044A FE9D 28 17 FE20 BMI WASON  
02045A FE9E 28 17 FE20 BMI WASON  
02046A FE9F 28 17 FE20 BMI WASON  
02047A FE9F 28 17 FE20 BMI WASON  
02048A FE9F 86 80 A DVTCLR LLA #580  
02049A FE9F 86 42 A RESTLP PSHS A,U  
02050A FE9F 86 42 A RESTLP PSHS A,U  
02051A FE9F 86 42 A RESTLP PSHS A,U  
02052A FE9F 86 42 A RESTLP PSHS A,U



# FM-7 BIOSソース・リスト



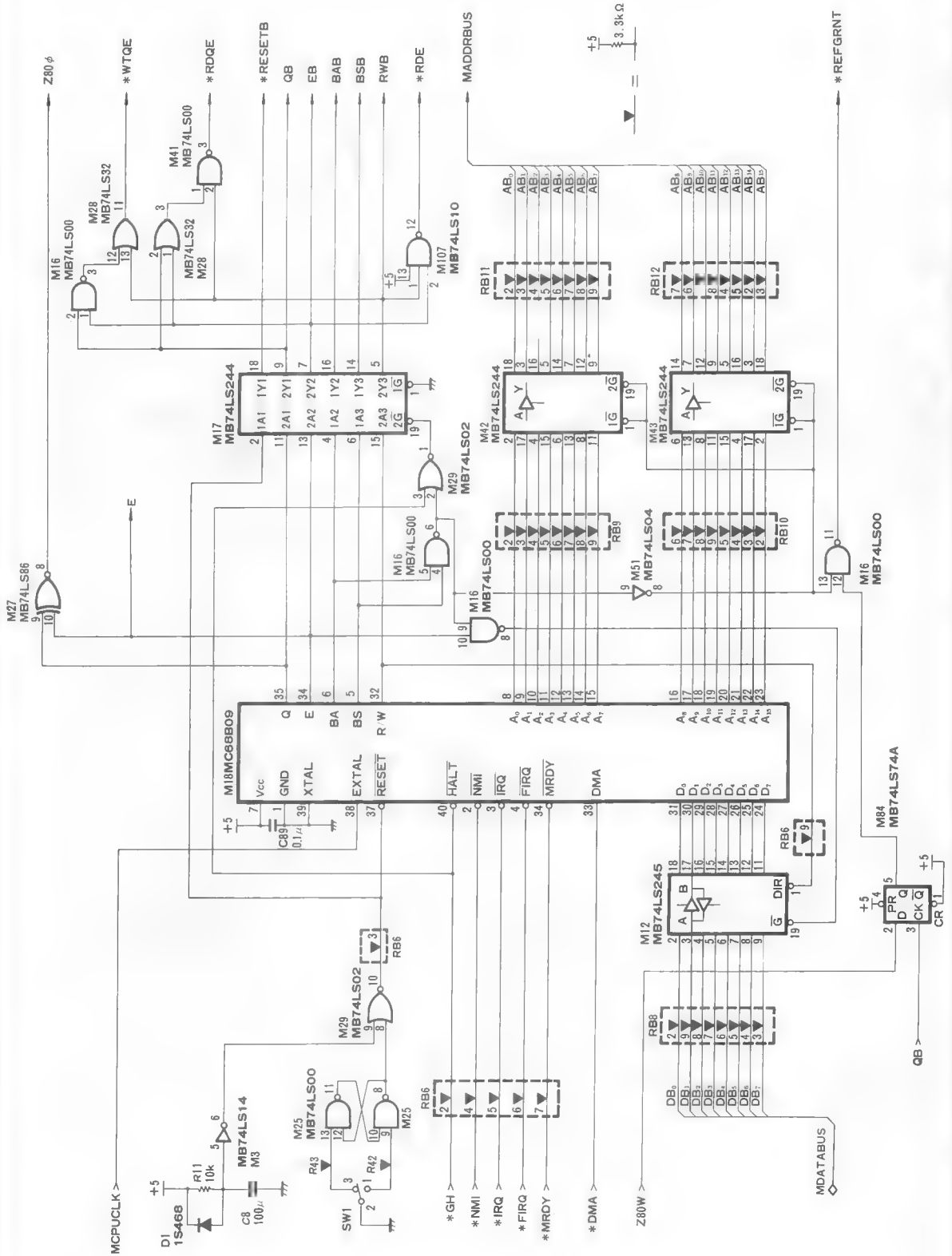
```

02053A FF13 81 84 A CMPA #84 ALL END ?
02054A FF15 26 F4 FF0B BNE RESTLP NO,CONTINUE.
02055A FF17 66 FF08 LDA DSTDRV
02056A FF1A 8A 80 A ORA #80
02057A FF1C 97 1D A STA DDRIE DESTINATION DRIVE SET.
02058A FF1E 20 09 FF23 BRA CHKRDY
02059 *
02060A FF20 B1 FFE0 A WASON CMPA CRDRIV DESTINATION=CURRENT ?
02061A FF23 27 04 FF23 BEQ CHKRDY YES
02062A FF25 8A 80 A ORA #80
02063A FF27 97 1D A STA DDRIE IT IS NOT SAME.DRIVE REG SET.
02064A FF29 8D 9F FECA CHKRDY BSR FREADY READY CHECK.
02065A FF2B 27 9A FEC7 BEQ CME30 NOT READY.
02066A FF2D 1C FE CLC NORMAL
02067A FF2F 39 RTS
02068 *
02069 FF30 A TM1S EQU * 1 SECOND TIMER=985
02070A FF30 CE 03D9 A LDU #03D9
02071A FF33 8D 09 FF3E TM1S5 BSR TM1M
02072A FF35 33 5F A LEAU -1,U
02073A FF37 1183 0000 A CMPU #0
02074A FF3B 26 F6 FF33 BNE TM1S5
02075A FF3D 39 RTS
02076 *
02077A FF3E 108E 00E0 A TM1M LDY #0E0 1 MS TIMER
02078A FF42 34 02 A PSHS A
02079A FF44 36 00 A LDA MTLPT
02080A FF46 47 ASRA
02081A FF47 25 04 FF4D BCS TM1M5
02082A FF49 108E 0099 A LDY #99
02083A FF4D 31 3F A TM1M5 LEAY -1,Y DEC Y
02084A FF4F 26 FC FF4D BNE TM1M5
02085A FF51 35 82 A PULS A,PC
02086 *
02087A FF53 34 04 A TM20M PSHS B
02088A FF55 C6 14 A LDB #20
02089A FF57 8D E5 FF3E TM20LP BSR TM1M

02090A FF59 5A DECB
02091A FF5A 26 FB FF57 BNE TM20LP
02092A FF5C 35 84 A PULS B,PC
02093 *****
02095 *
02096 FF5E A RDWRIT EQU * SECTOR WRITE.
02097 FF5E A RREAD EQU *
02098A FF5E 8D FE55 A JSR SEEK DRIVE SELECT ,SEEK TRACK
02099A FF61 26 5A FF6D BNE DREAD7
02100 *
02101A FF63 1A50 A FDB MSKFIQ IRQ,FIRQ MASK.(82/01/20).
02102 *
02103A FF65 A6 06 A LDA RCBSID,X
02104A FF67 84 01 A ANDA #1 GET UNIT.
02105A FF69 37 1C A STA DSIDE
02106A FF6B A6 05 A LDA RCBSCT,X
02107A FF6D 37 1A A STA DSCTR SECTOR SET
02108A FF6F A6 84 A LDA RQND,X GET REQUEST NO.
02109A FF71 81 0A A CMPA #FOREAD DREAD ?
02110A FF73 27 18 FF8D BEQ DREADX YES ,BRANCH DREAD.
02111A FF75 86 A0 A LDA #8A0 WRITE COMMAND SET
02112A FF77 37 18 A STA DCMO
02113A FF79 10AE 02 A LDY RCBD8A,X
02114A FF7C A6 A0 A DWRIT1 LDA 0,Y+ DATA GET
02115A FF7E D6 1F A DWRIT2 LDB DDRQ DRQ ?
02116A FF80 2A 04 FF86 BPL DWRIT5 NO
02117A FF82 37 18 A STA DDATA DATA STORE
02118A FF84 20 F6 FF7C BRA DWRIT1
02119 *
02120A FF88 58 * DWRIT5 ASLB EXTRACT COMMAND END TO MSB.
02121A FF87 2A F5 FF7E BPL DWRIT2
02122A FF89 8D B3 FF3E BSR TM1M
02123A FF8B 20 14 FFA1 BRA RWEND
02124 *
02125 *****
02126 *
02127 FF8D A DREADX EQU * SECTOR READ.

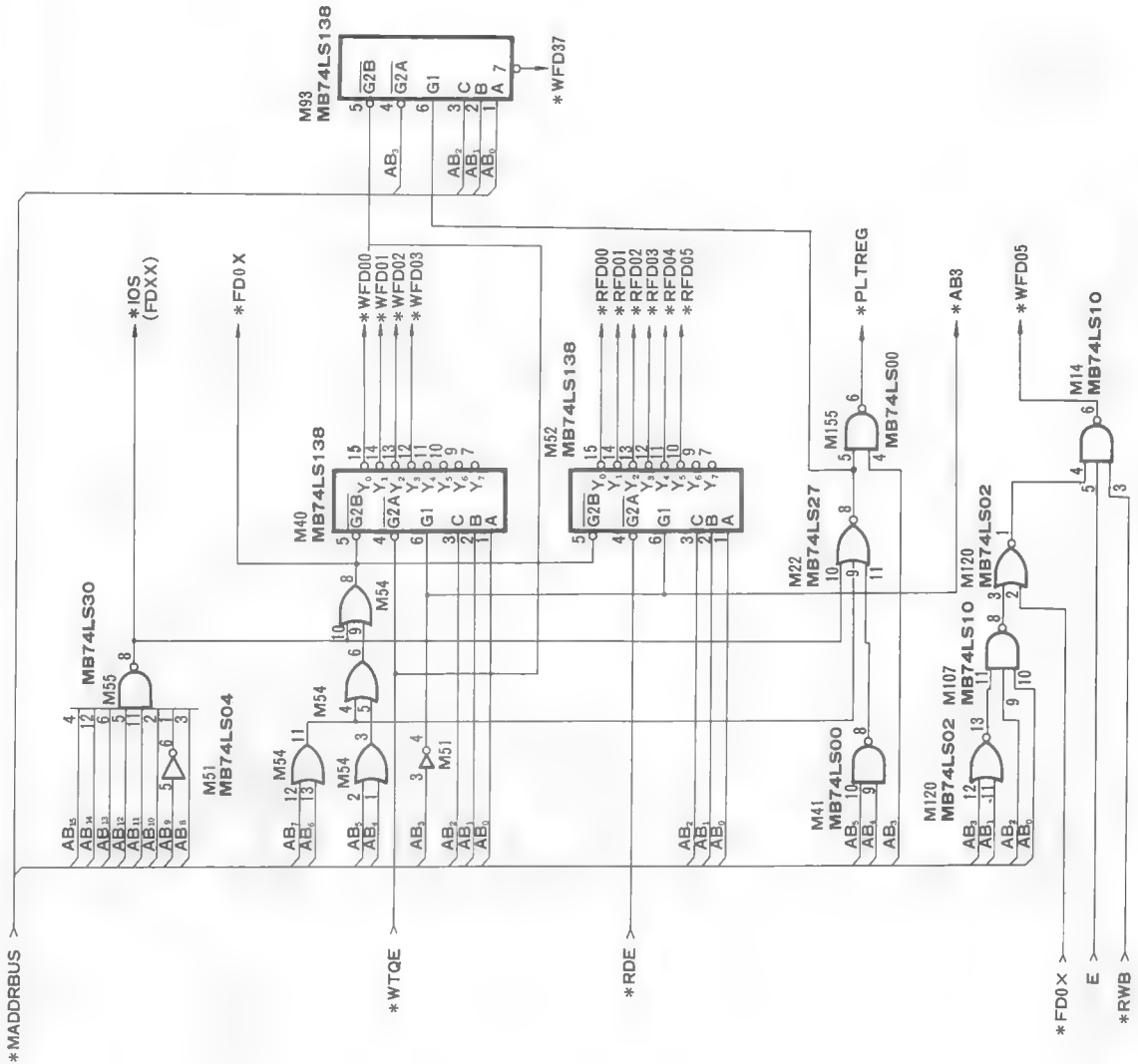
02128A FF8D 86 80 A LDA #80 READ COMMAND SET
02129A FF8F 97 18 A STA DCMO
02130A FF91 10AE 02 A LDY RCBD8A,X
02131A FF94 D6 1F A DREAD1 LDB DDRQ DRQ ?
02132A FF96 2A 06 FF9E BPL DREAD5 NO
02133A FF98 36 18 A LDA DDATA LOAD DATA
02134A FF9A A7 A0 A STA 0,Y+
02135A FF9C 20 F6 FF94 BRA DREAD1
02136 *
02137A FF9E 58 * DREAD5 ASLB EXTRACT COMMAND END TO MSB.
02138A FF9F 2A F3 FF94 BPL DREAD1
02139A FFA1 D6 18 A RWEND DSTAT ERROR ?
02140A FFA3 26 02 FFA7 BNE DREAD6
02141A FFA5 4F CLR A
02142A FFA8 39 RTS
02143 *
02144A FFA7 86 0A A DREAD6 LDA #0A
02145A FFA9 C5 80 A BITB #80 NOT READY ?
02146A FFA8 26 10 FFBD BNE DREAD7
02147A FFA0 4C INCA
02148A FFAE C5 40 A BITB #40 WRITE PROTECTED ?
02149A FFB0 26 08 FFBD BNE DREAD7
02150A FFB2 4C INCA
02151A FFB3 C5 14 A BITB #14 RECORD NOT FOUND,LOST DATA ?
02152A FFB5 26 06 FFBD BNE DREAD7
02153A FFB7 4C INCA
02154A FFB8 C5 08 A BITB #08
02155A FFB8 26 01 FFBD BNE DREAD7
02156A FFB9 4C INCA
02157A FFB0 1A 01 DREAD7 SEC $0E DELETED DATA MARK DETECTED !
02158A FFBF 33 RTS SET CARRY FOR ORSEL ERROR INTERFACE.
02159 *
02160 * * BOOT ROM END ADDRESS=$FFDF
02161 *
02162A FFFE ORG $FFFE
02163A FFFE FDB BOTENT BOOT ENTRY ADDRESS.
02164 END
    
```

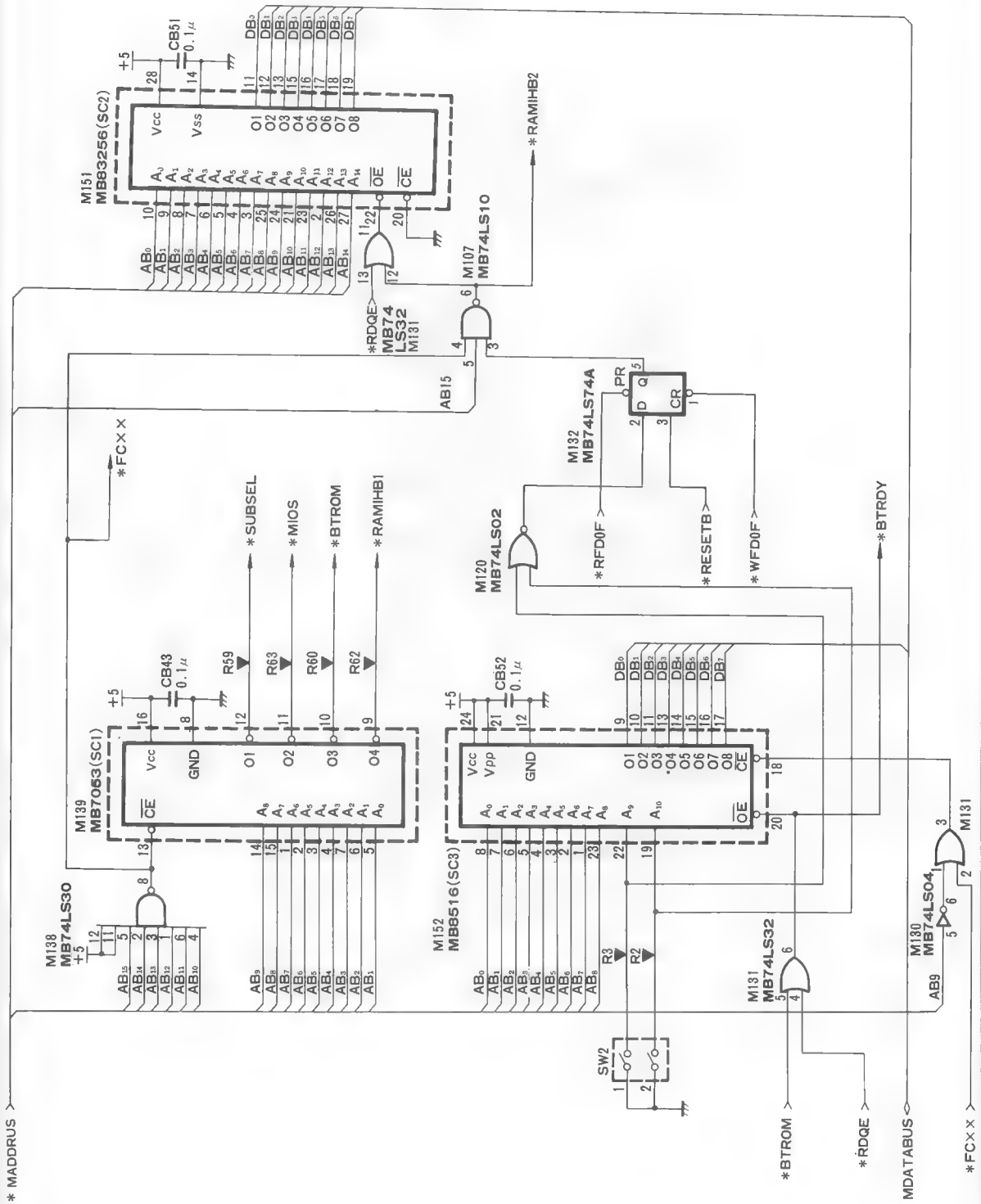
## メインCPU部



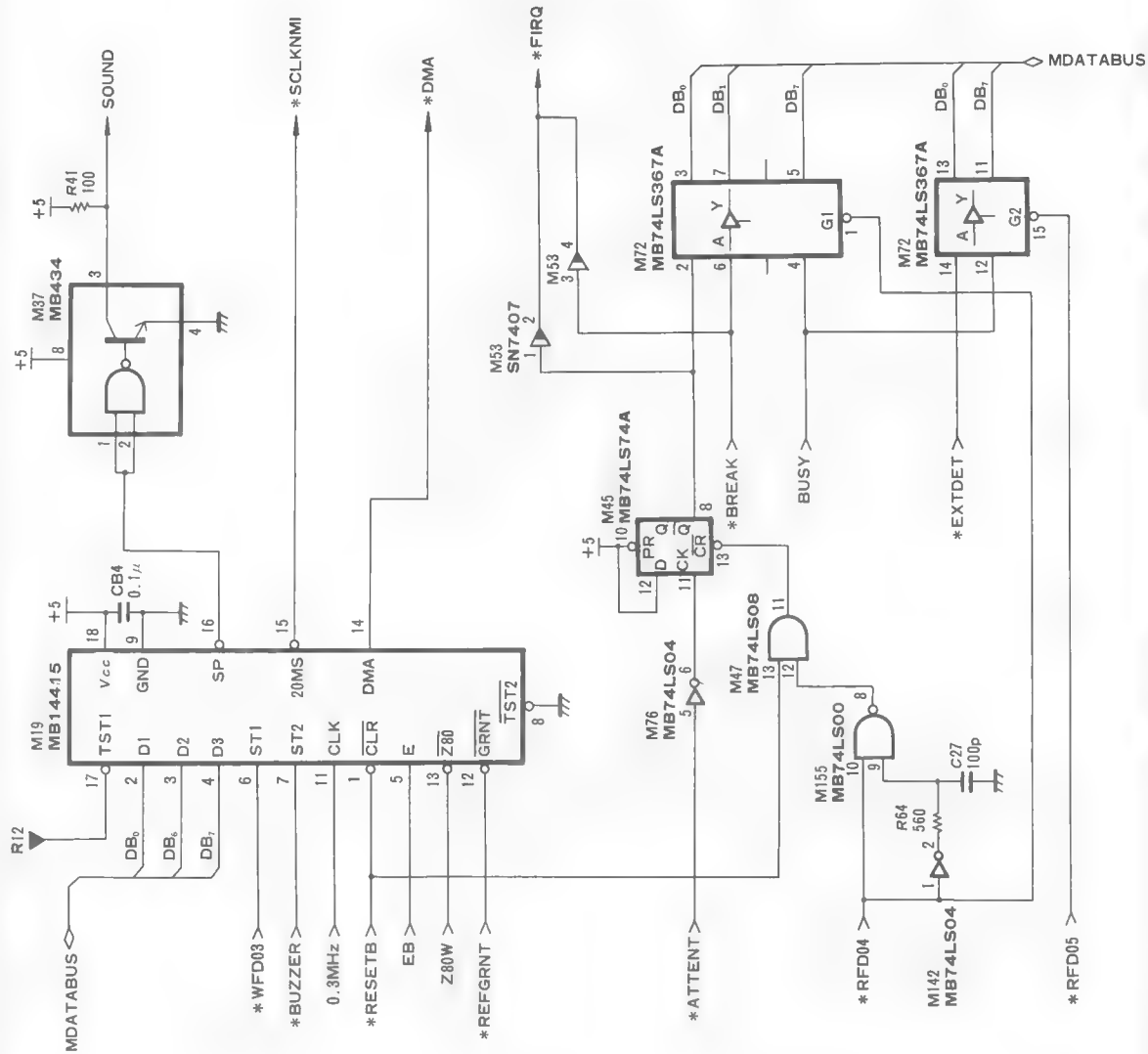


## メインI/Oアドレス部



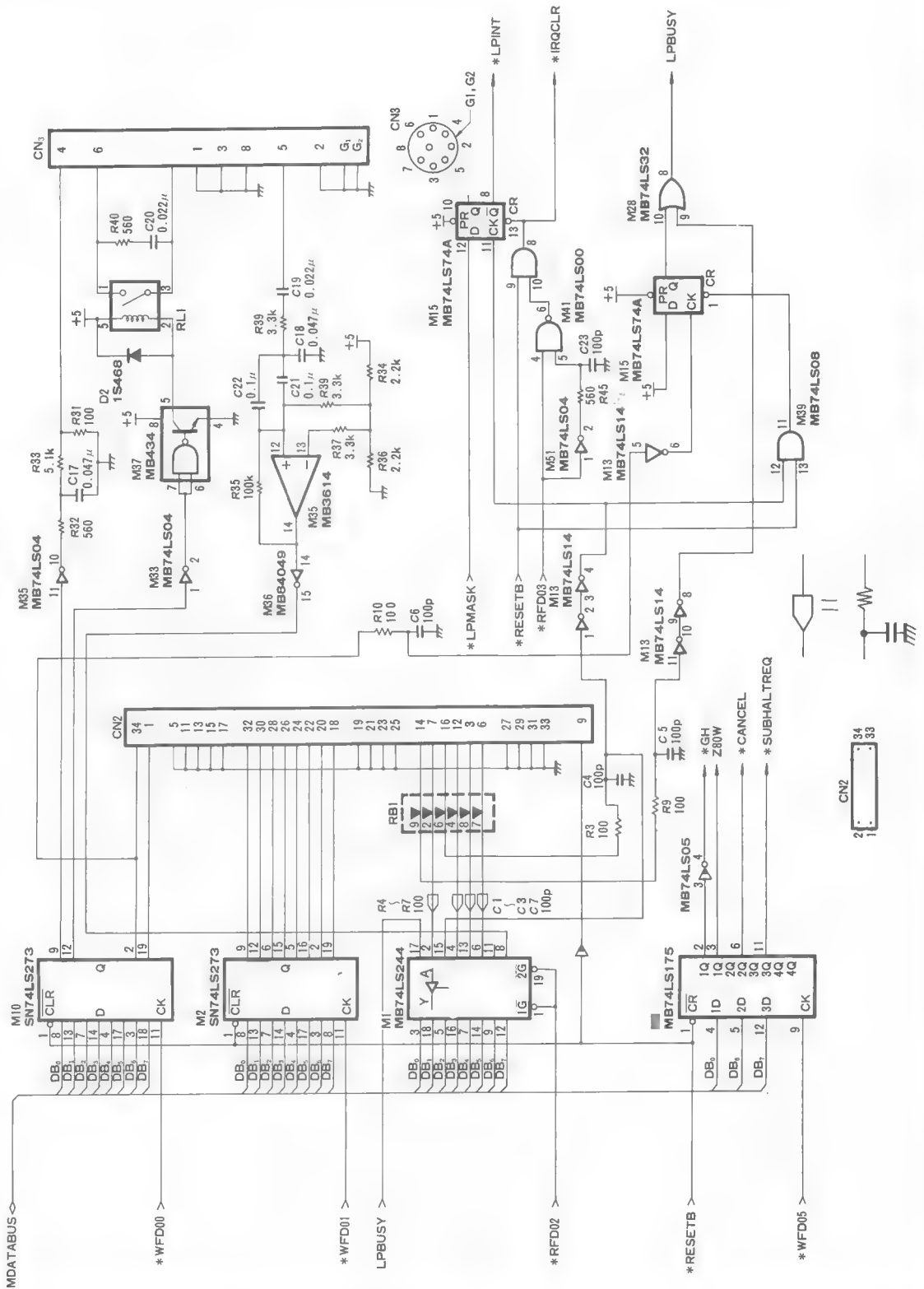




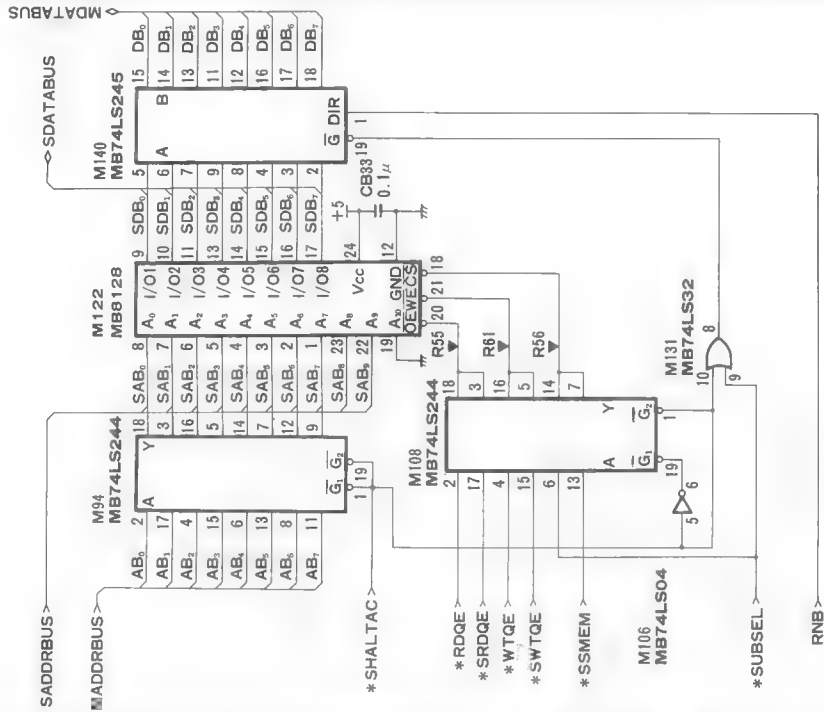




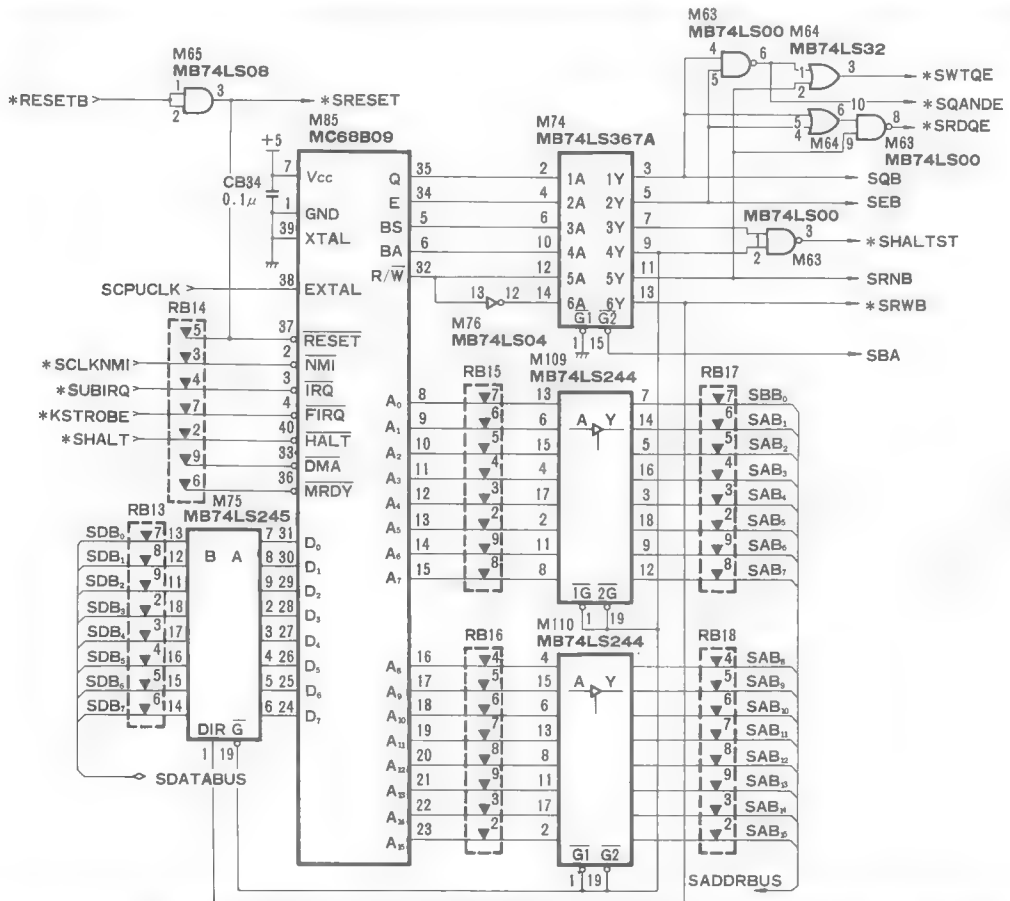
メイン プリント・カセット部



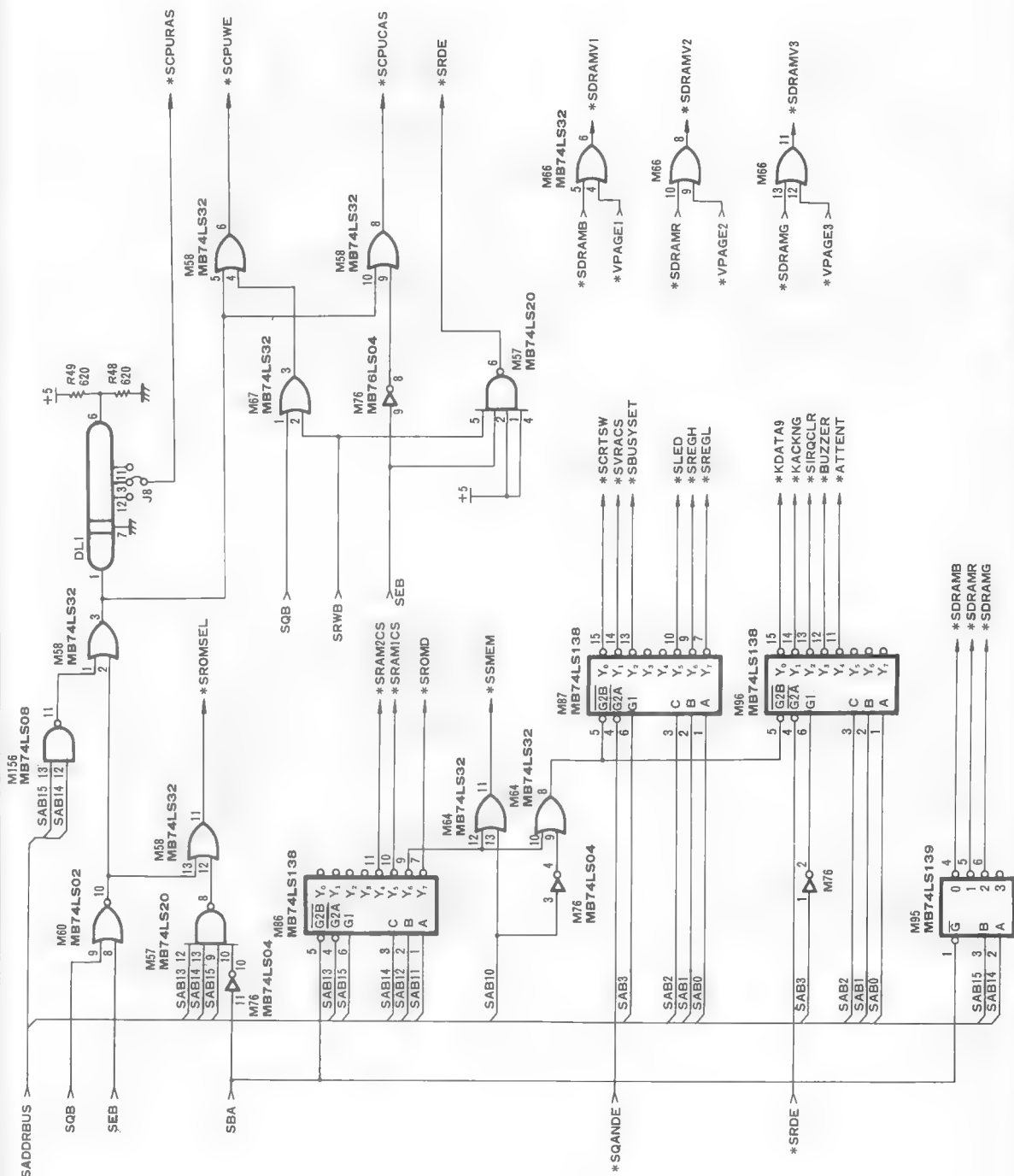
共有メモリ部

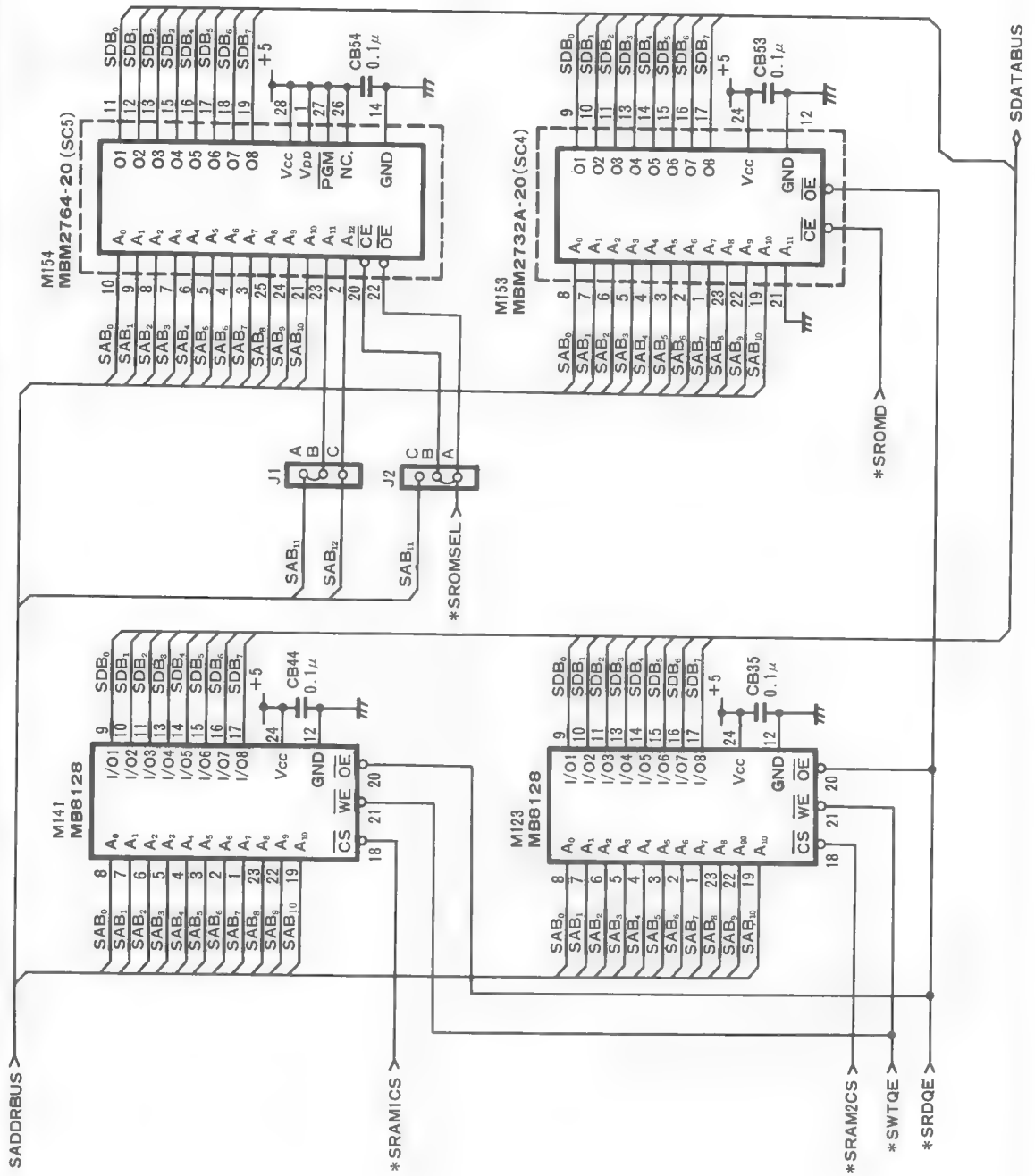


サブCPU部

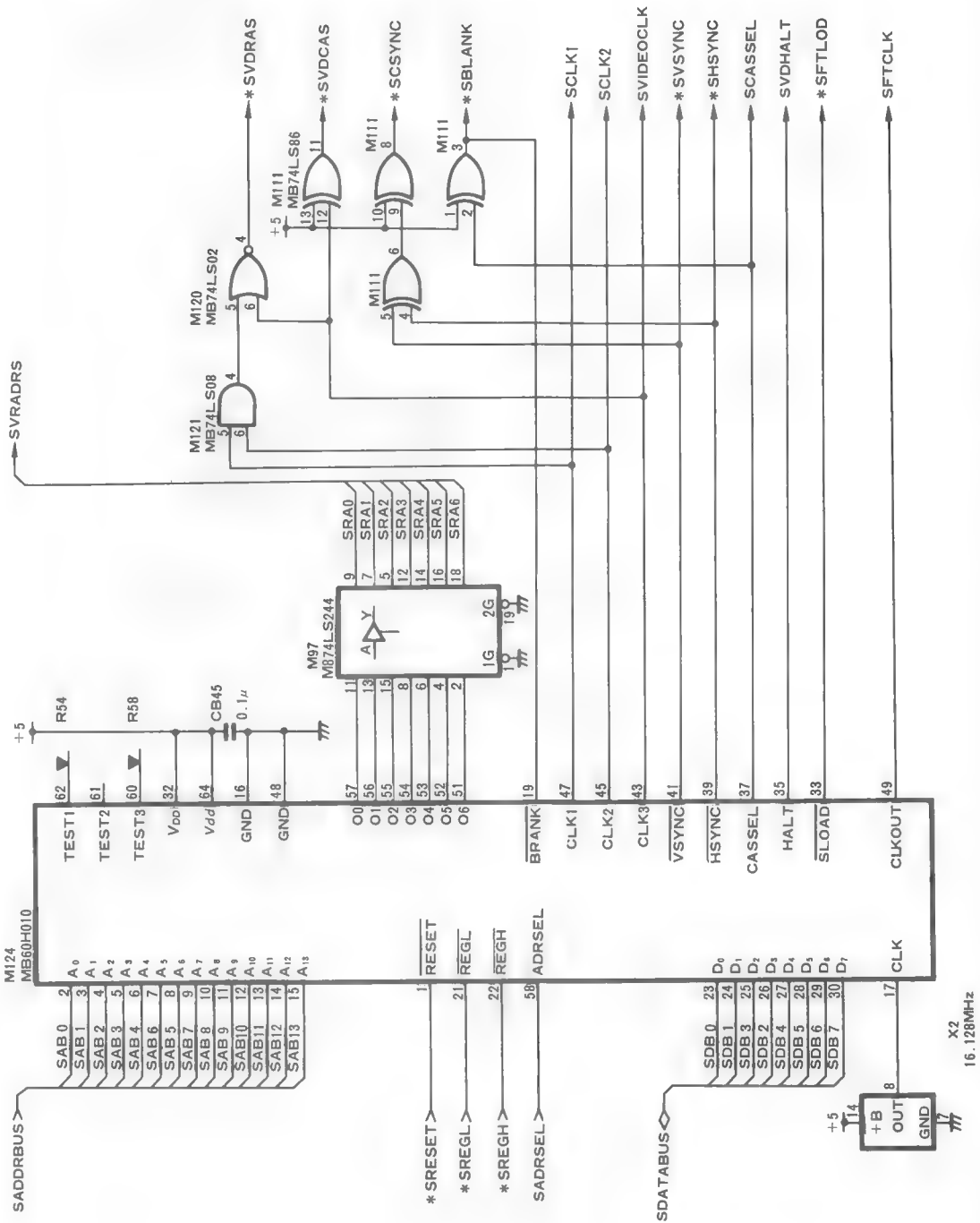


サブ アドレス・デコード部

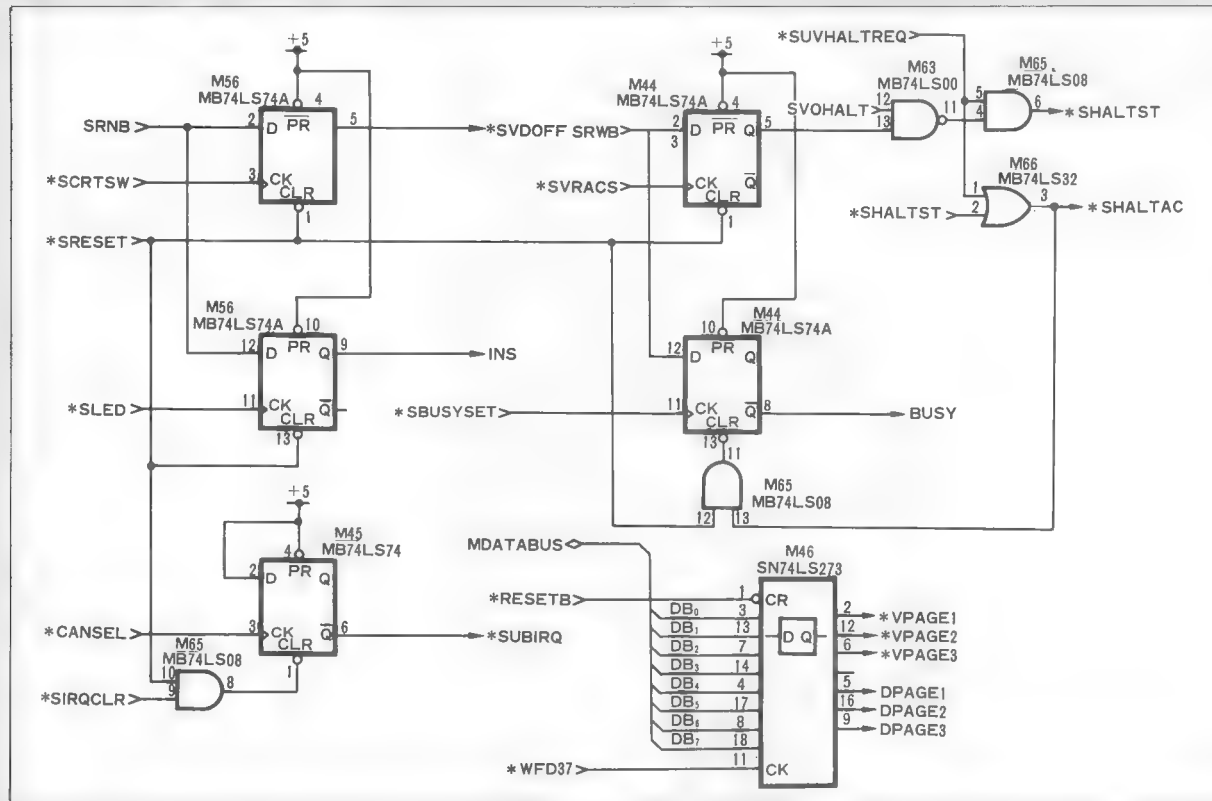




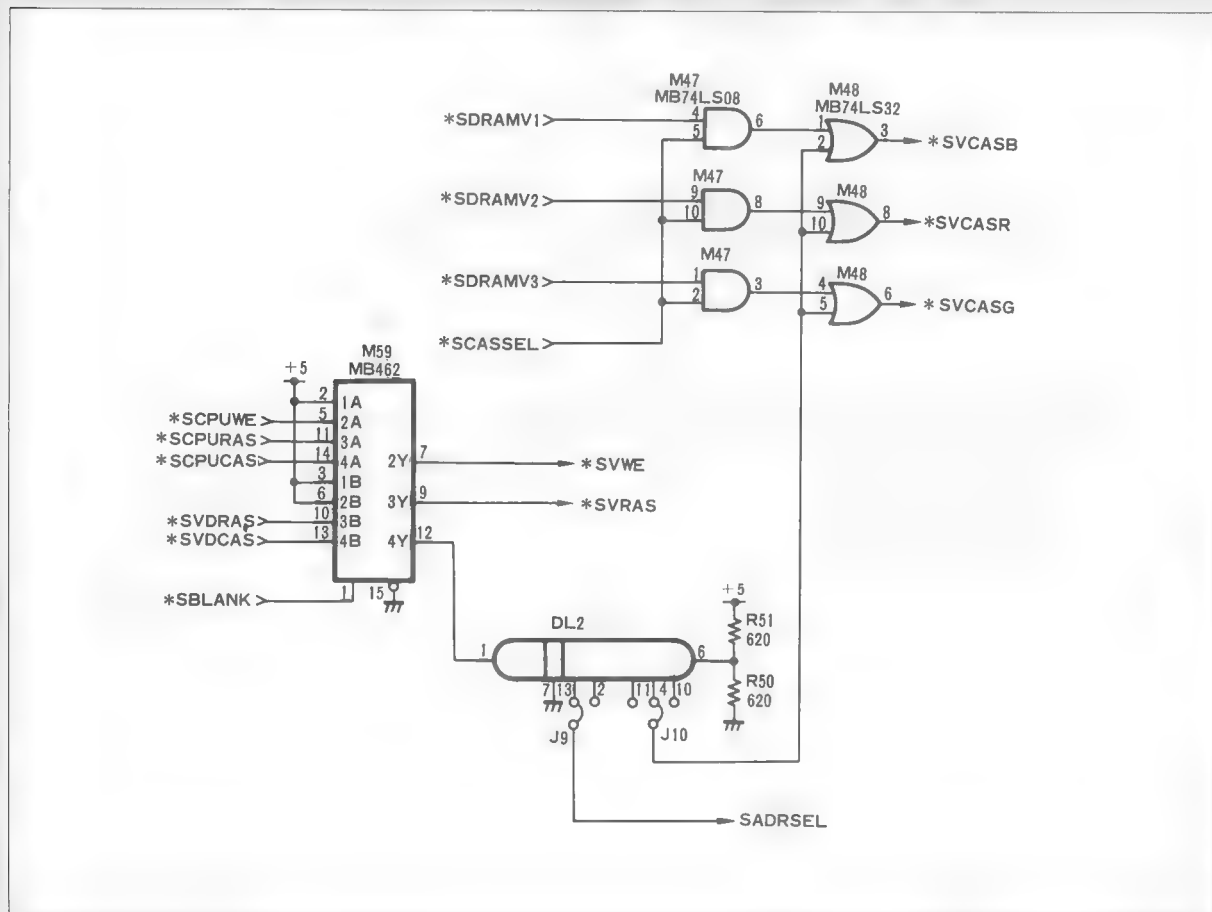
サブ CRT CNTRL部



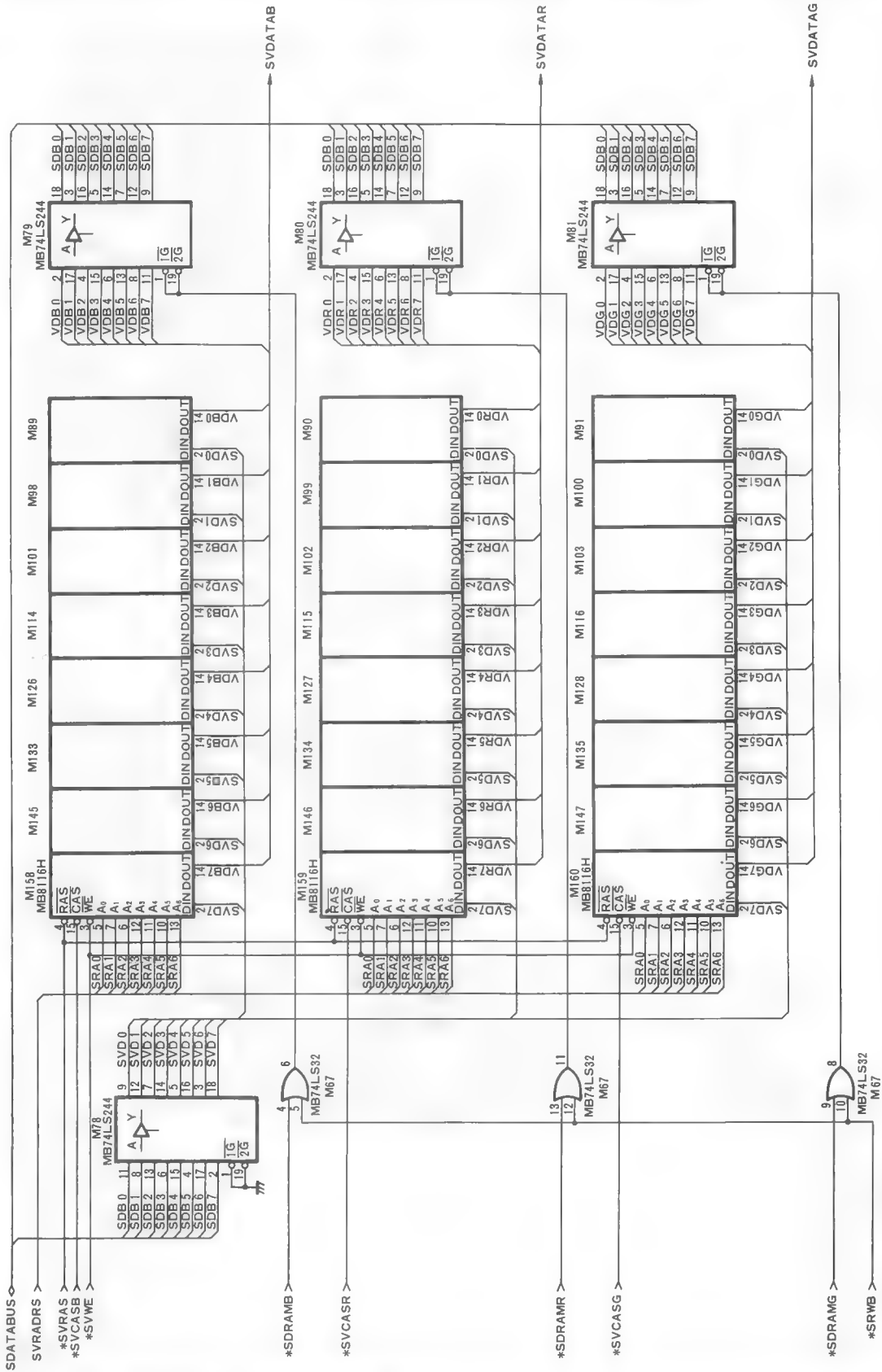
## サブ REG/FLAG部



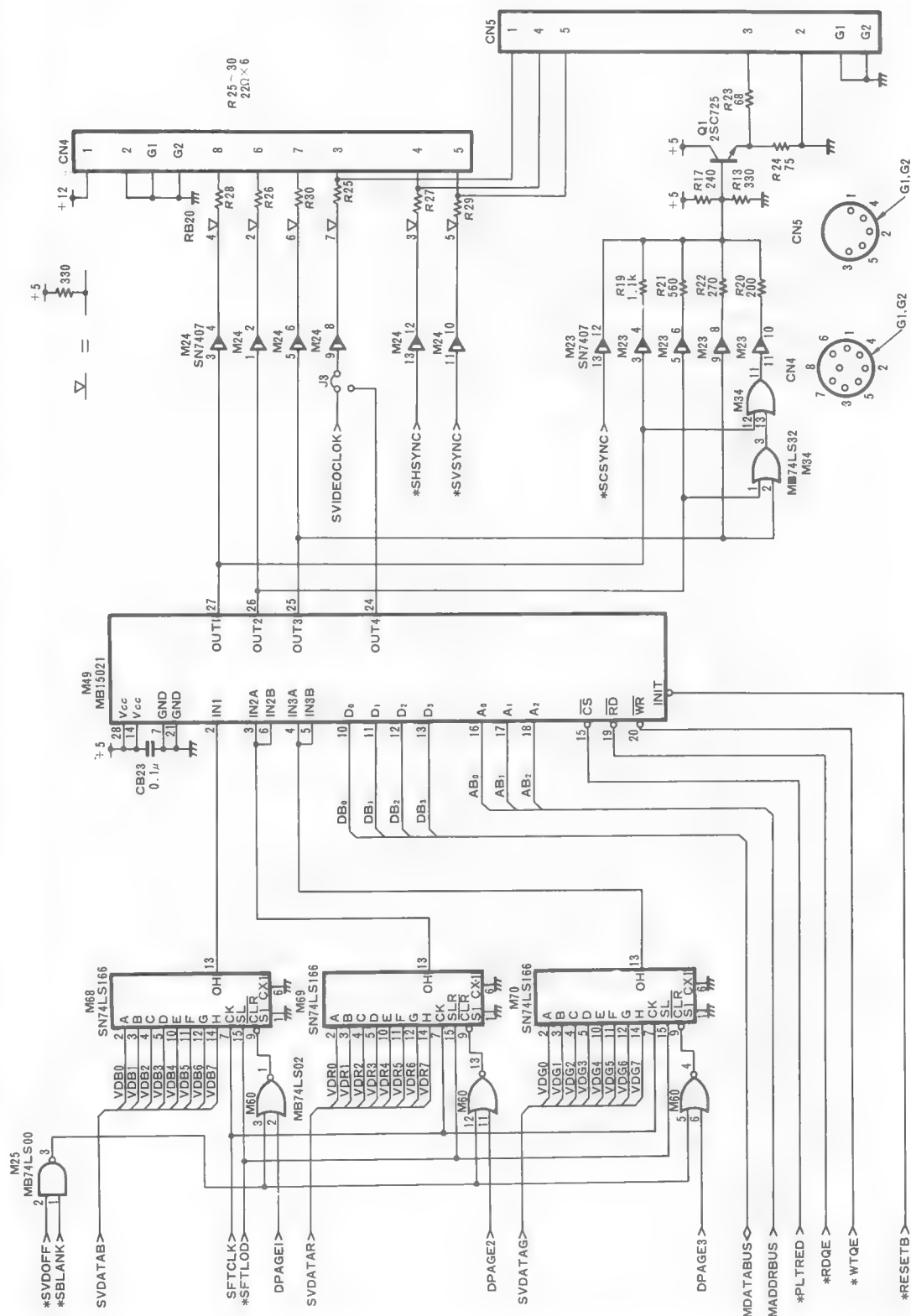
## サブ CRTアドレス部



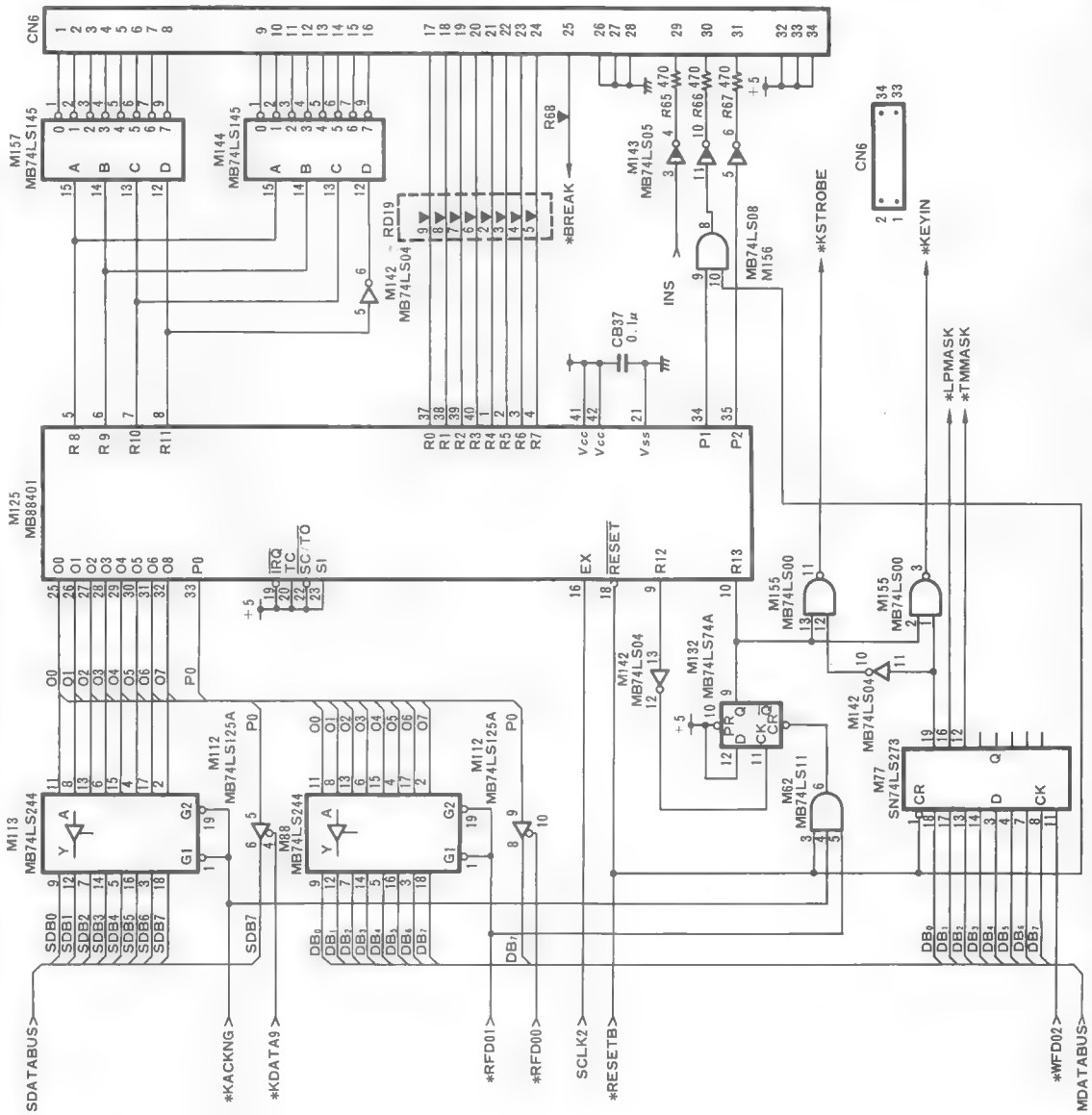
サブ CRT RAM部

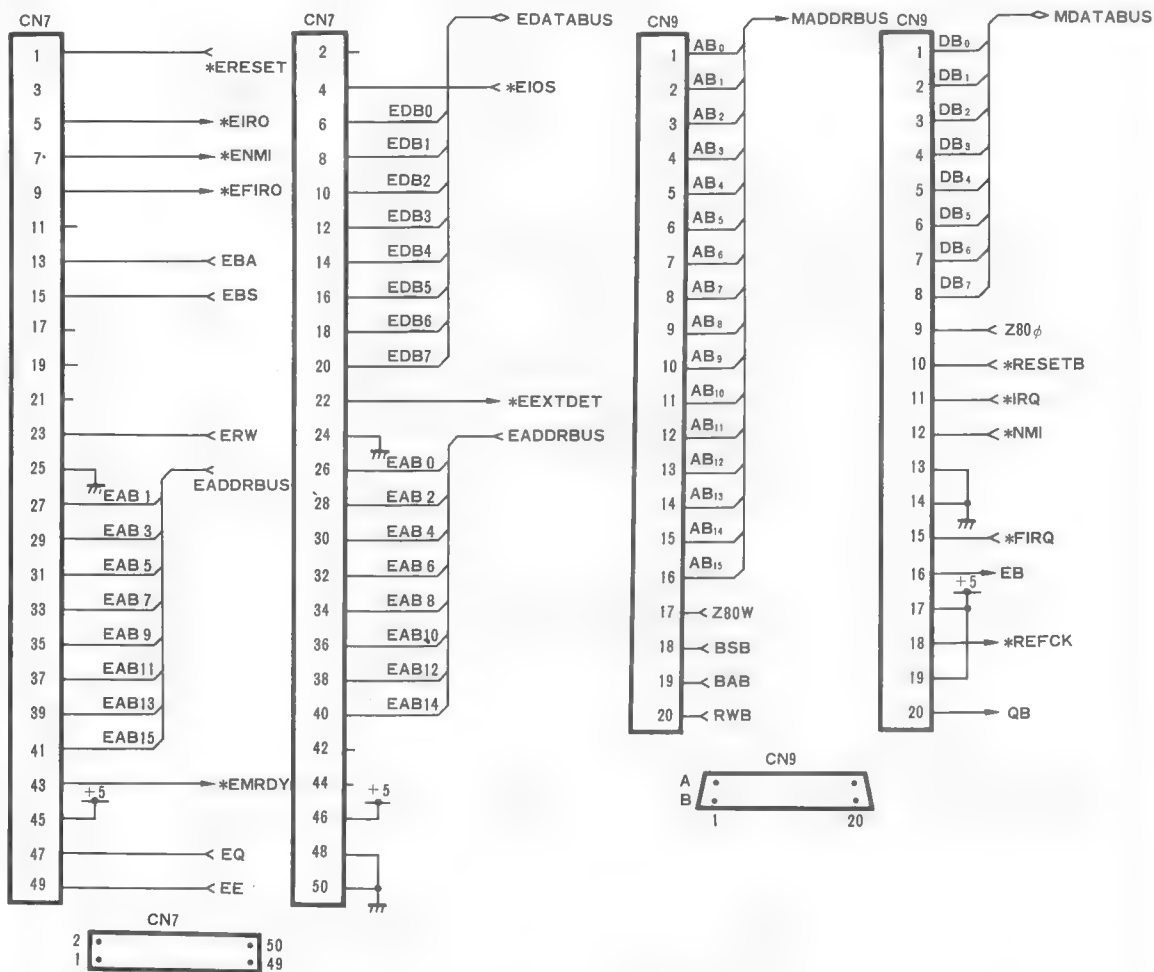




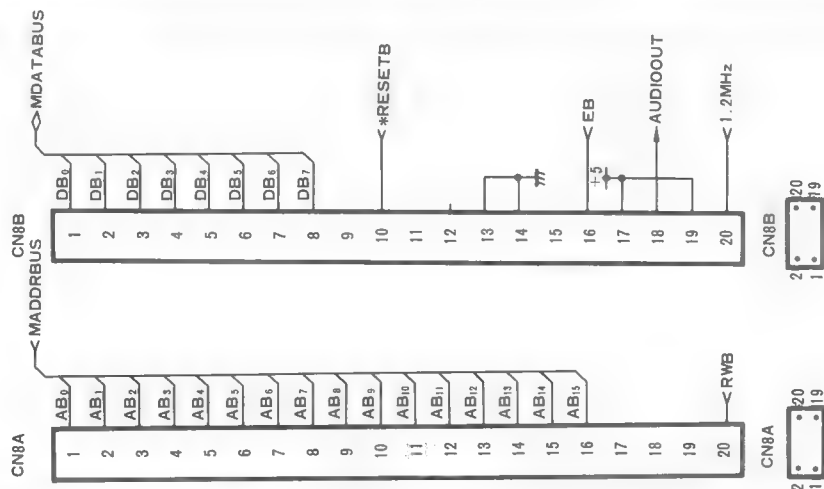


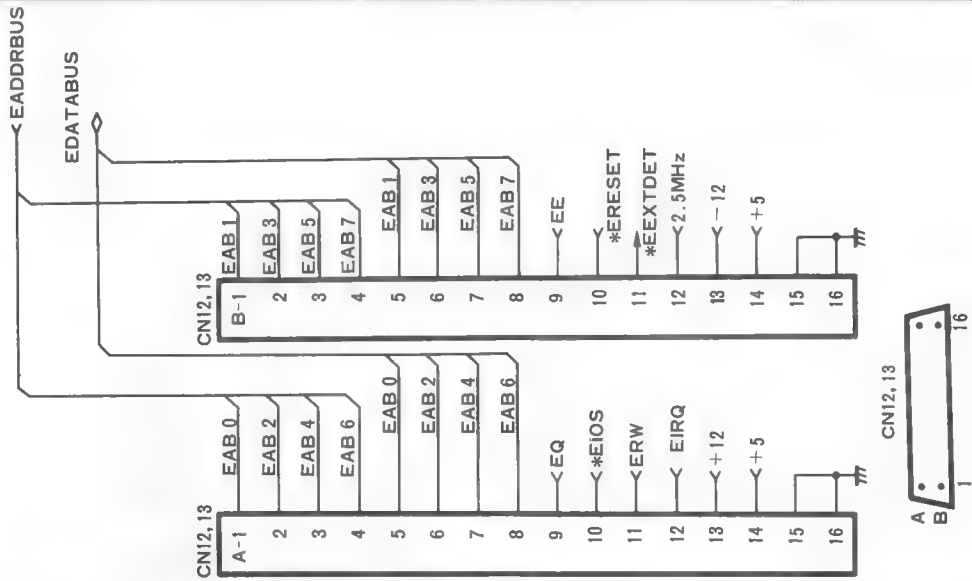
キーインターフェイス部



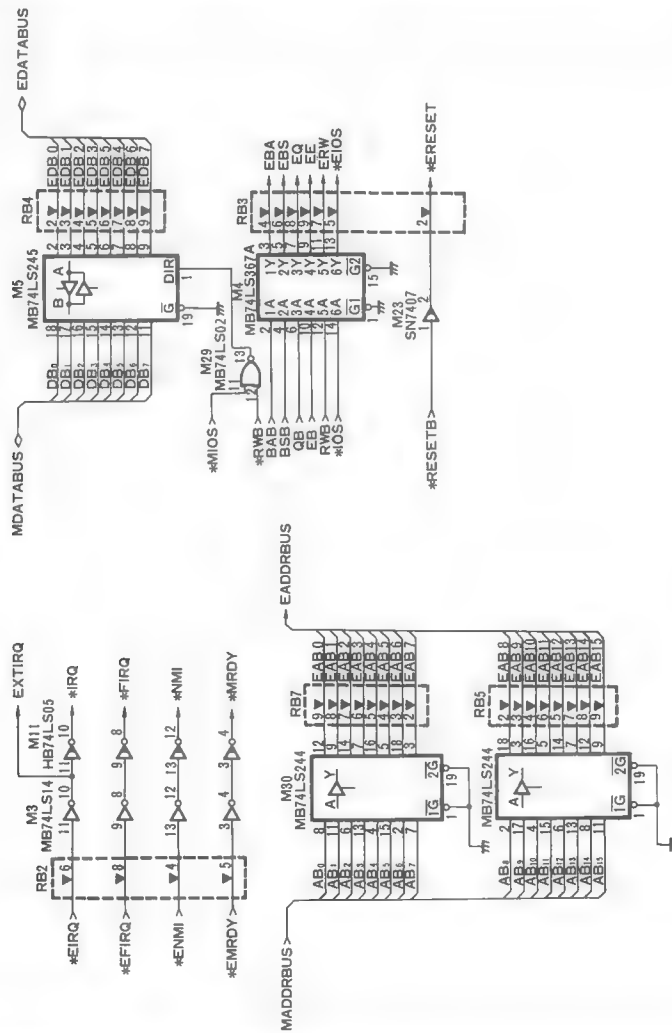


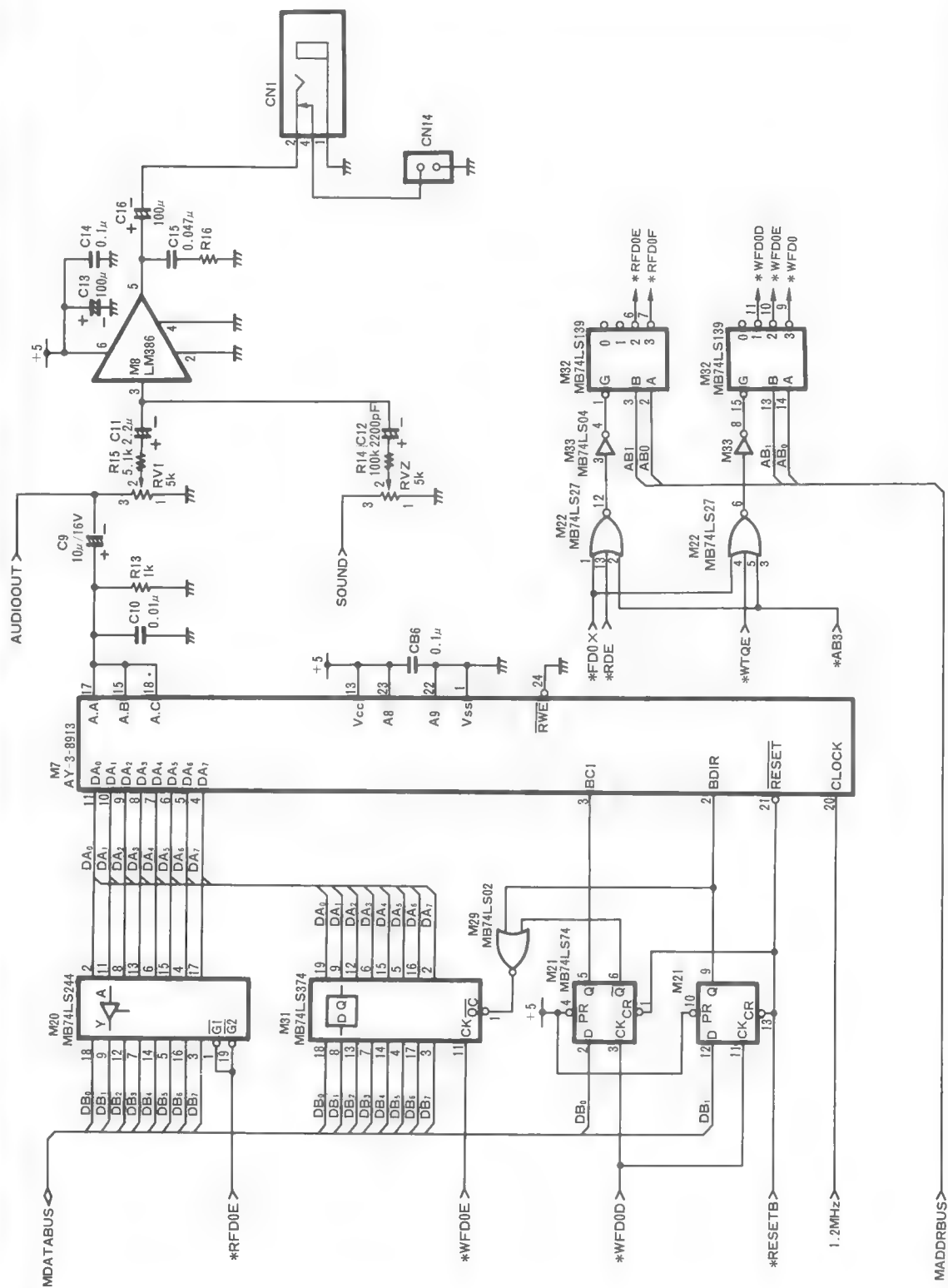
コネクタ PSG部

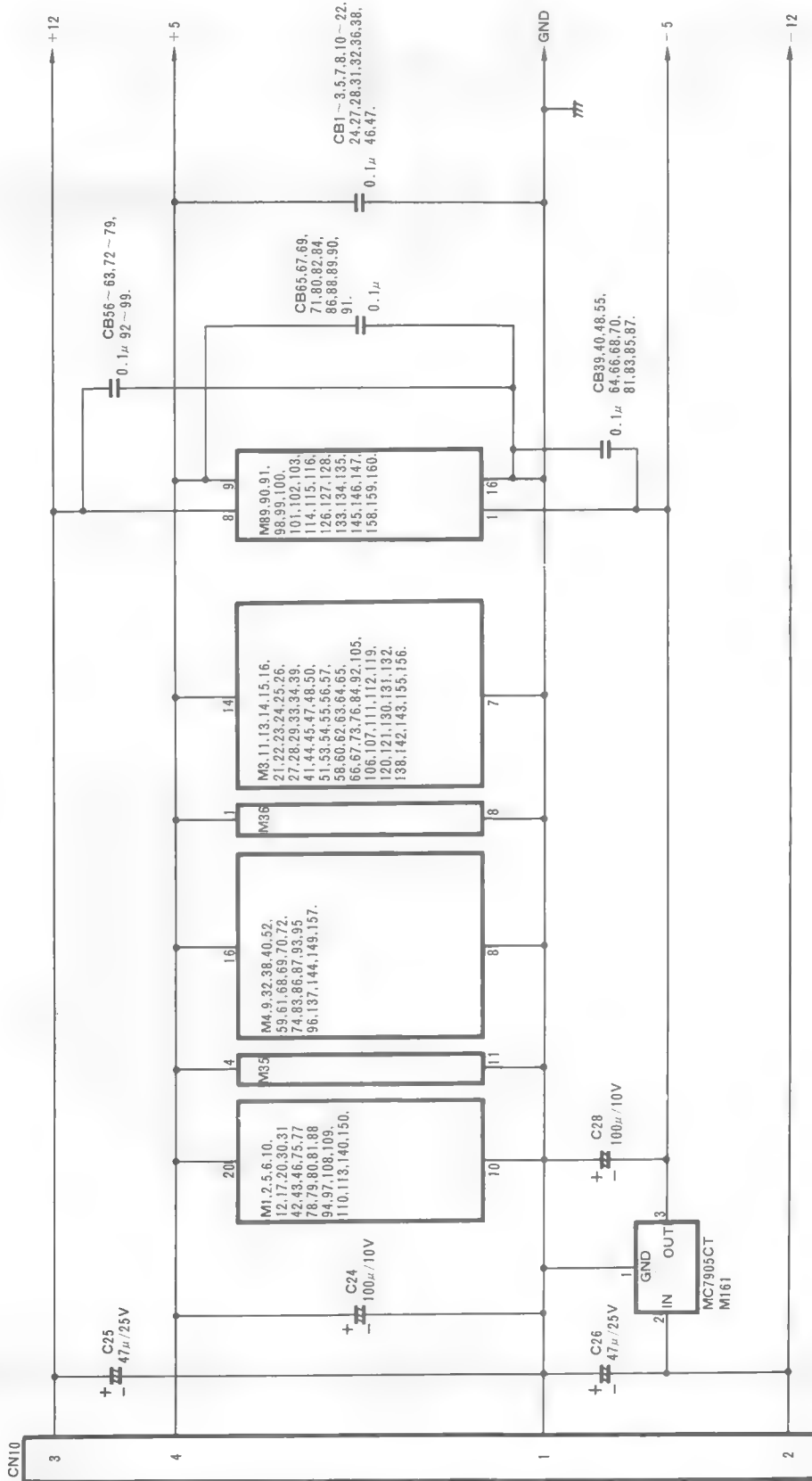




## 拡張バス・バッファ部







# バブル・モード とDOSモード

## 概要

FM-8のブートROM(MB8516)は図1のとおり4分割されて使われています。メイン・メモリ上にどのBANKが現われるかは、本体背面にあるディップ・スイッチ(図2)の9番、10番の設定によります。

FM-8のディップ・スイッチの設定とメモリ・マップの関係を図3に示します。

表1からもわかるとおり、ディップ・スイッチの切り替えにより、F-BASIC ROM/RAM、ブートROMのBANKの両方が切り替わります。

バブル・モード、DOSモードとはそれぞれ、ブートROM BANK1、BANK2がメイン・メモリ・マップにエントリするモードであり、\$0000~\$FC7Fは連続RAM領域になります。これらのモードにおいては、システム・プログラム(OS, F-BASIC V2.0, etc)はすべて外部記憶装置よりメモリ上にローディングされなければなりません。

システム・プログラムをバブル・カセットからローディングする場合はバブル・モードであり、フロッピーディスクからローディングする場合はDOSモードです。この呼び方は便宜的なものであり、正しくはミニFDモードと呼ぶべきかもしれません。

## ■システムの起動

FM-8におけるシステムの起動は、図4に示すとおり3段階で行なわれます。ブートROM内にはブート・ストラップ・ローダが格納されており、外部記憶にはIPLとシステム・プログラムが格納されています。ブート・ストラップ・ローダはIPLプログラムを読み出し、IPLプログラムはシステム・プログラムを読み出し、システム・プログラムにコントロールが渡され、システムが起動します。

ブート・ストラップ・ローダは固定的なプログラムであり、様々なシステムに対応するのはシステムごとに作成さ

れるIPLプログラムです。IPLプログラムはその記憶される場所、バイト数が固定的に定められています。

- ①バブル・モード
  - ユニット0のバブル・カセット
  - ページ0~ページ15の16ページ
  - 512バイト(32×16バイト)
- ②DOSモード
  - ユニット0のディスケット
  - トラック0, サイド0, セクタ1~2,
  - 512バイト(256×2バイト)

ブート・ストラップ・ローダは上記の場所よりIPLプログラムを読み出し、\$300~\$4FFに格納し、\$300番地に分岐します。以上がバブル・モードとDOSモードの概要です。

図3 FM-8メモリ・モデル

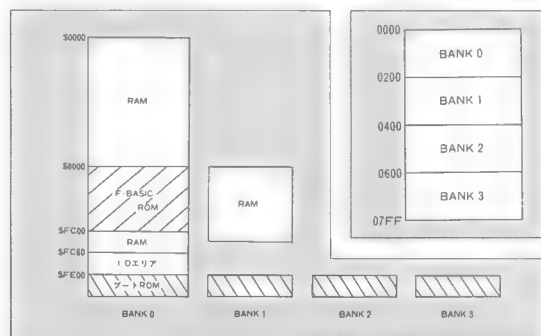


図1 ブートROMのBANKアドレス

図2 ディップ・スイッチの設定



表1

ディップ 9	ディップ 10	\$8000~\$FBFF メモリ属性	\$FE00~\$FFDF ブートROM BANK No.	有 効	注 意
ON	ON	ROM	0	ROMモード	F-BASIC V1.0の起動
ON	OFF	RAM	1	バブル・モード	バブル・カセットのシステム・プログラムの起動
OFF	ON	RAM	2	5' DOSモード	F-BASIC V2.0(5')やミニFDOSの起動
OFF	OFF	RAM	3	8' DOSモード	F-BASIC V2.0(8')や標準FDOSの起動

## バブル・モード, DOSモードにおけるIPLプログラムの作成

IPLプログラム作成のポイントは次の3つです。

- ① ディスクまたはバブル上でのシステム・プログラムの格納場所
- ② メイン・メモリ上でのシステム・プログラムの格納場所
- ③ (ディスクの場合) ブートROM内のディスク・アクセス・ルーチンの利用

バブルのアクセス・ルーチンは特に用意されていませんが、以下に示すバブルのブート・ストラップ・ロードのリストを参照すれば比較的簡単に作成することができます。

①, ②はシステムごとに定まってくることであり, ③はFM-8のハードウェアに依存します。フロッピーディスクのアクセス・ルーチンは少々複雑ですが, ブートROMの中に用意されています。

ブートROMのルーチンの使用法は、基本的にBIOSコールの方法と同一です。しかしながら、BIOSの正式なエントリ・ポイントからのエントリではないので、IPLプログラムにおいて次のことを追加して行なわなければなりません。

- ① レジスタのセーブ (破壊されては困るレジスタを保存する)
- ② DPレジスタを\$FDにセットする。

上記のことはディスクの場合のことですので、バブルの場合はアクセス・ルーチンをすべて作成しなければなりません。

ブートROMのディスクのアクセス・ルーチンのエントリ・アドレスは次のとおりです。

\$FE02	リストA・ルーチン
\$FE05	セクタ・ライト・ルーチン
\$FE08	セクタ・リード・ルーチン

ディスクのIPLプログラムのフローチャートは図5の通りです。

では、バブル・モードとDOSモードのIPLプログラムの実例をリスト1, 2に示します。どちらの場合もF-BASICを起動するものです。

## 最後に

IPLプログラム、システム・プログラムができ上がったとき、それをバブルまたはディスクに書き込まなければなりません。それは専用のユーティリティを作成して行なうか、またはBIOSコールにより、物理的にページ・ライトまたはセクタ・ライトすることにより行なえます。

以上に、バブル・モードとDOSモードについて概要および実例を示しました。しかし、実際にIPLプログラムを作成して、リセットからシステムを立ち上げようとする場合はIPLプログラムを詳細にトレースし、動作を理解することが肝要です。

そこで、バブル・モードとDOSモードのブートROMのアセンブル・リストをリスト3, 4に示します。また、8" DOSモードのリストをリスト5に示しますので参考にしてください。なお、ROMモードについては『FM-8活用研究』のBIOSのリストを見てください。

図4 システムの起動

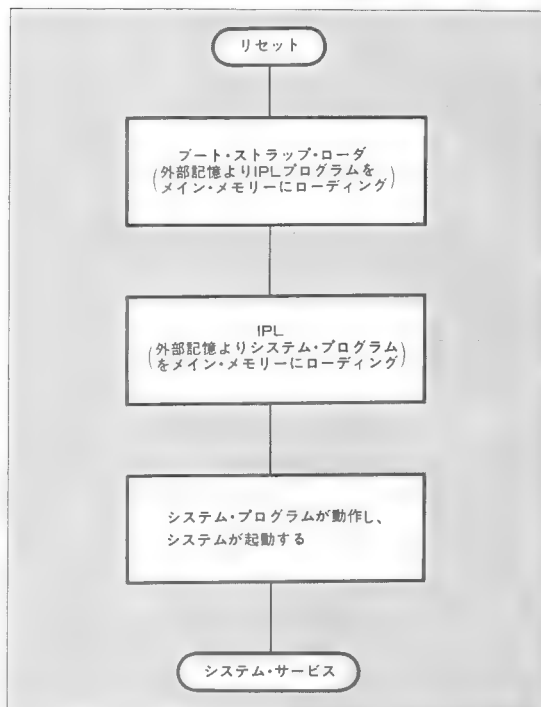
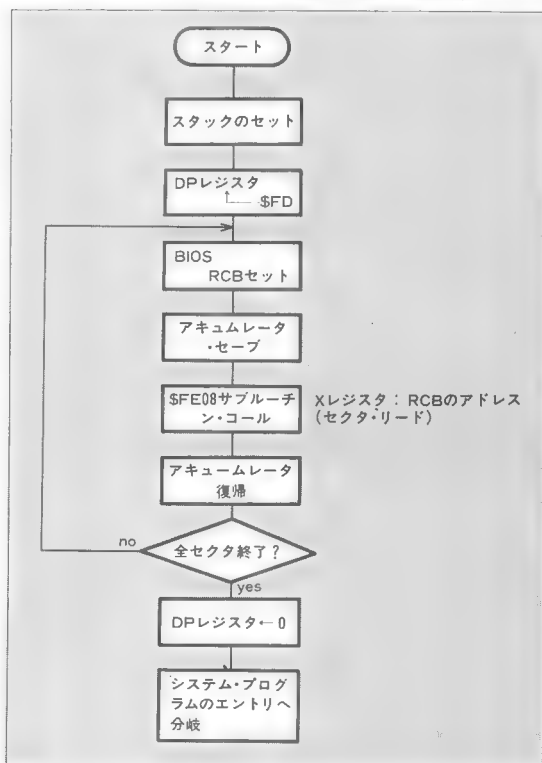


図5 IPLプログラムのフローチャート





## リスト 1 バブル・モードの IPL 例 (バブルより F-BASIC を起動)

```

                                TTL      Bubble BASIC IPL program Ver 1.0
                                OPT      SYM,NOGEN
                                ORG      $0300  IPL program start

0300
*
* IPL program by N.K.
*
0003 BUZZER EQU $03      Buzzer status Reg.
0005 HALT EQU $05      Sub-system halt flag.
0010 BDATA EQU $10     Data Reg.
0011 BCMD EQU $11     Command Reg.
0012 BSTAT EQU $12     Status Reg.
0014 BPGADH EQU $14    Page address Reg.
0016 BPGCTH EQU $16    Page count Reg.
*
* Program start
*
0300 0300 START EQU *
0300 86 FD LDA #$FD
0302 1F 88 TFR A,DP      DPR is $FD
0304 8E FFFF LDX $FFFF
0307 BUSY EQU *
0307 30 1F LEAX -1,X    Over load?
0309 27 35 0340 BEQ BBLERR Yes, bubble error!
0308 96 12 LDA BSTAT    Bubble ready?
030D 44 LSRA
030E 25 F7 0307 BCS BUSY No,now busy.
0310 CE 8000 LDU $8000 BASIC loading start address.
0313 8E 001F LDX #31    Bubble loading start page.
0316 9F 14 STX BPGADH   Store to address Reg.
0318 8E 03E2 LDX #994   Bubble loading pages.
0318 9F 16 STX BPGCTH   Store to count Reg.
031D 86 01 LDA #1       Bubble read command set.
031F 97 11 STA BCMD
0321 READ10 EQU *
0321 96 12 LDA BSTAT    Status Reg. load.
0323 28 0A 032F BMI READ20 Command finished?
0325 85 20 BITA #$20    Can read?
0327 27 F8 0321 BEQ READ10 No,can't.
0329 96 10 LDA BDATA    Load data Reg.
032B A7 C0 STA ,U+      Store data & address up.
032D 20 F2 0321 BRA READ10
032F 96 12 LDA BSTAT    Command finished?
0331 2A EE 0321 BPL READ10 No. Continue!
0333 85 20 BITA #$20    Any error?
0335 28 08 0340 BNE BBLERR Yes error.
0337 1C AF ANDCC #$AF
0339 4F CLRA
033A 1F 88 TFR A,DP      DPR clear.
033C 8E 9F FBFE JMP [$FBFE] Jump to BASIC hot start.
*
* Print error message
*
0340 86 81 0340 BBLERR EQU *
0342 97 03 LDA #$81
0344 20 FE 0344 LOOP STA BUZZER Buzzer on.
                                BRA LOOP No end loop.
                                0300 END START

```

## リスト 2 DOS モードの IPL 例 (5" ディスクより F-BASIC を起動)

```

                                NAM      BASIPL
                                TTL      IPL OF 40K RAM MODE SYSTEM
                                OPT      NOMEM,OBJ,NOG
                                ORG      $300
0300 0300 20 18 031A LENTRY BRA START
*
0094 DCSIZE EQU 148     DISK CODE SIZE.
6800 DCSADR EQU $6800   DISK CODE START ADDRESS.
0302 IPLRCB EQU *
0302 0A FCB 10          READ SECTOR.
0303 00 FCB 0           STATUS.
0304 6800 FCB $6800     DISK CODE START ADDRESS.
0306 02 FCB 2           DISK CODE START TRACK.
0307 01 FCB 1           DISK CODE START SECTOR.
0308 00 FCB 0           DISK CODE START SIDE.
0309 00 FCB 0           UNIT 0.
*
030A 030A IDRCB EQU *   ID READ.
030A 0A FCB 10,0
030C 03A1 FCB IDBUF
030E 00 FCB 0,3,0,0 TRACK 0. SECTOR 3. SIDE 0.
*

```

リスト2 DOSモードのIPL例(5"ディスクよりF-BASICを起動)

0312	08		RESRCB	FCB	8	RESTORE.
0313	0006			RMB	6	
0319	00			FCB	0	UNIT 0.
*						
031A	10CE	0300	START	LDS	##300	
031E	86	FD		LDA	##FD	
0320	1F	8B		TFR	A,DP	
0322	8E	030A		LDX	#IDRCB	
0325	8D	51	0378	BSR	DCREAD	READ ID
0327	AE	02		LDX	2,X	
0329	EC	84		LDD	,X	
032B	1083	5359		CMPD	##S*256+'Y SYSTEM ?	
032F	26	68	0399	BNE	IPLERR	NO.
0331	A6	02		LDA	2,X	
0333	81	53		CMPA	##S	
0335	26	62	0399	BNE	IPLERR	NO.
0337	C6	94		LDB	#DCSIZE	YES.
0339	8E	0302		LDX	#IPLRCB	READ DISK CODE.
033C	34	04	RDLOOP	PSHS	B	
033E	BD	FE08		JSR	FDREAD	READ DISK CODE.
0341	4D			TSTA		
0342	27	06	034A	BEQ	REDOK	
0344	81	0A		CMPA	#10	DRIVE NOT READY ?
0346	27	51	0399	BEQ	IPLERR	YES.
0348	20	2E	0378	BRA	DCREAD	RETRY.
034A	CC	0100	REDOK	LDD	##100	
034D	E3	02		ADD	2,X	
034F	ED	02		STD	2,X	RENEW RAM ADDRESS.
0351	A6	05		LDA	5,X	
0353	4C			INCA		
0354	81	10		CMPA	#16	CURRENT SECTOR+1 > 16 ?
0356	2F	0D	0365	BLE	NEWSCT	NO.
0358	6D	06		TST	6,X	YES, SIDE 0 ?
035A	27	05	0361	BEQ	SIDINC	YES.
035C	6C	04		INC	4,X	RENEW TRACK NO.
035E	6F	06		CLR	6,X	CHANGE OF SIDE *SIDE=0
0360		8C		FCB	##C	CMPX #,SKIP INC 6,X
0361	6C	06	SIDINC	INC	6,X	CHANGE OF SIDE :SIDE=1
0363	86	01		LDA	#1	
0365	A7	05	NEWSCT	STA	5,X	RENEW SECTOR NO.
0367	35	04		PULS	B	
0369	5A			DECB		
036A	26	D0	033C	BNE	RDLOOP	
*						
036C	4F			CLRA		
036D	1F	8B		TFR	A,DP	
036F	8E	6800		LDX	#DCSADR	
0372	86	FF		LDA	##FF	
0374	6E	9F FBFE		JMP	[BASENT]	
*						
0378	C6	05	DCREAD	LDB	#5	
037A	34	04	RETRY	PSHS	B	
037C	34	10		PSHS	X	
037E	8E	0312		LDX	##RESRCB	
0381	BD	FE02		JSR	RESTOR	
0384	35	10		PULS	X	
0386	81	0A		CMPA	#10	DRIVE NOT READY ?
0388	27	0F	0399	BEQ	IPLERR	YES.
038A	BD	FE08		JSR	FDREAD	
038D	35	04		PULS	B	
038F	4D			TSTA		
0390	27	0E	03A0	BEQ	RETRTS	
0392	81	0A		CMPA	#10	DRIVE NOT READY ?
0394	27	03	0399	BEQ	IPLERR	YES.
0396	5A			DECB		
0397	26	E1	037A	BNE	RETRY	
*						
0399	86	81		IPLERR	LDA	##81
039B	B7	FD03		STA	BELL	
039E	20	FE	039E	BRA	*	
*						
03A0	39			RETRTS	RTS	
*						
				* SYSTEM FLAG		
	FD03			BELL	EQU	##FD03
	FBFE			BASENT	EQU	##FBFE
	FE08			FDREAD	EQU	##FE08
	FE02			RESTOR	EQU	##FE02
*						
	03A1			IDBUF	EQU	*
	0300			END		LENTY

## リスト 3 バブル・モード(BANK 1)アセンブル・リスト

```

*
*      TTL      5 INCH BUBBLE MODE(NO 1) BOOT STRAP LOADER (B2 03 01).
*
*
*      ORG $FE00-----FORMAL EXECUTION.
*      ORG $EE00-----DEBUGING.
*
FE00
FE00 20  FE00 09  FE0B  A  BOTENT  EQU  *  $FE00  FM-8 BOOT PROGRAM BUBBLE MODE.(NO 1)
                        EQU  BRA  *  STARTS
                        *
FE07F  A  BOTSTK  EQU  *  $FC7F  STACK POINTER SET.
0300  A  IFLADR  EQU  *  $0300
*
FE02 7E  FE5B  A  RESTOR  JMP  RRESTR
FE05 7E  FF61  A  DWRITE  JMP  RDWRIT
FE0B 7E  FF61  A  DREAD  JMP  RDREAD
*
FE0B CC  FFC3  A  STARTS  EQU  *
FE0E FD  FFFC  A  LDD  *NMIRTI NMI VECTOR SET.
FE11 86  FD  A  STD  PNMI
FE13 1F  BB  A  LDA  *%FD
FE15 8E  FFFF  A  LDX  *%FFFF  DIRECT PAGE=FD..
FE1B 30  1F  A  TIMDLY  LEAX  -1,X  ABOUT 0.5 SEC DILAY.
FE1A 26  FC  FE1B  A  BNE  TIMDLY
FE1C 8E  FFFF  A  LDX  *%FFFF
FE1F 30  1F  A  BUSY  LEAX  -1,X
FE21 27  32  FE55  BEQ  BBLERR
FE23 96  12  A  LDA  BSTAT
FE25 44
FE26 25  F7  FE1F  BCS  BUSY
FE2B CE  0300  A  LDU  *$0300
FE2B 8E  0000  A  LDX  #0
FE2E 9F  14  A  LOOP  STX  BPGADH
FE30 8E  0010  A  LDX  #16
FE33 9F  16  A  STX  BPGCTH
FE35 86  01  A  LDA  #1
FE37 97  11  A  STA  BCMD
FE39 96  12  A  READ10  LDA  BSTAT
FE3B 2B  0A  FE47  BMI  READ20
FE3D 85  20  A  BITA  *$20
FE3F 27  F8  FE39  BEQ  READ10
FE41 96  10  A  LDA  BDATA
FE43 A7  C0  A  STA  ,U+
FE45 20  F2  FE39  BRA  READ10
FE47 96  12  A  READ20  LDA  BSTAT  CME ?
FE49 2A  EE  FE39  BPL  READ10
FE4B 85  02  A  BITA  *$02  ERROR ?
FE4D 26  06  FE55  BNE  BBLERR
FE4F 4F
FE50 1F  BB  A  TFR  A,DP
FE52 7E  0300  A  JMP  IFLADR
*
FE55 86  B1  A  BBLERR  EQU  *
FE57 97  03  A  LDA  *$B1
FE59 20  FE  FE59  BRA  *
*
*      TTL      5 INCH FD DRIVER MODULE(F-BASIC : ROM-BASED MODE.)
*
*
*-----*
*
*      ***** TYPE 1 COMMAND
*
FE5B CE  FE5B  A  RRESTR  EQU  *  SEEK TRACK 0.
FE5E A6  FFE1  A  LDU  *  #CURTRK
07  A  LDA  RCOUNT,X
* BUG CORRECTION 2 --> ZS VERSION B2/01/11 SOURCE UPDATE.
FE60 B4  03  A  RRESEN  EQU  *  RRESEN LABEL POSITION WAS CORRECTED.CLR A,U -> ANDA *$03.
FE62 6F  C6  A  ANDA  *$03  UNIT SELECT.
FE64 D6  1D  A  CLR  A,U  TRACK ADDRESS 0 SET.
FE66 BA  80  A  LDB  DDRIVE
FE68 97  1D  A  ORA  *$80
FE6A 8D  4C  FE6B  STA  DDRIVE
FE6C 96  1D  A  BSR  CME15
FE6E 2A  4D  FE6D  LDA  DDRIVE
FE70 5D  4D  BPL  CME16
FE71 2B  0B  FE7B  TSTB
FE73 B6  02  A  BMI  Z1000
FE75 BD  FF3F  A  LDA  #2
FE7B 4A  A  JSR  TMIS
FE79 26  FA  FE75  DECA  Z500
FE7B B6  0A  A  BNE  Z500
FE7D 20  1A  FE99  LDA  *$0A  RESTORE COMMAND (V=0)
                        BRA  CMEXEC  BRANCH COMMAND EXECUTION
*
FE7F BD  7B  FE7F  A  SEEK  EQU  *  SEEK TRACK
FE81 25  3A  FE8D  BSR  DRSEL  DRIVE SELECT
FE83 CE  FFE1  A  BCS  CME16
FE86 F6  FFE8  A  LDU  *CURTRK
FE89 A6  C5  A  LDB  DSTDRV
FE8B 97  19  A  LDA  B,U
FE8D A6  04  A  STA  DTRACK
FE8F A1  C5  A  LDA  RCBTRK,X DESTINATION TRACK LOAD.
FE91 27  24  A  CMPA  B,U
FE93 A7  C5  FE97  BEQ  CME11  NON SEEK ACTION
FE95 97  1B  A  STA  B,U  NEW TRACK ADDRESS SET.
FE97 86  1E  A  DDATA  TRACK NO IN DATA REG
                        LDA  *$1E  SEEK COMMAND SET (V=1)

```

## リスト 3 バブル・モード(BANK 1)アセンブル・リスト

```

*
FE99 97 FE99 A CMEXEC EQU * COMMAND EXECUTION
FE9B 86 18 A STA CMD COMMAND STORE
FE9B 86 03 A LDA #3

*
FE9D 4A CME02 DECA TIME OVER ?
FE9E 27 31 FED1 BEQ CME25 YES, TIME OVER.
FEA0 10BE CB00 A CME05 LDY ##CB00 ABOUT 1.4 SECOND
FEA4 31 3F A CME10 LEAY -1,Y
FEA6 27 F5 FE9D BEQ CME02

*
FEA8 D6 1F A LDB DDR0 READ DR0,IR0
FEA8 C5 40 A BITB ##40 IR0REG ?
FEAC 27 F6 FEA4 BEQ CME10 BUSY, LOOP
FEAE D6 18 A LDB DSTAT MAKE SURE

*
FEB0 BD FF56 A JSR TM20M

*
FEB3 C5 18 A BITB ##18 CRC, SEEK ERROR ?
FEB5 26 08 FEBF BNE CME20 ERROR OCCURED
FEB7 4F CME11 CLRA
FEB8 BD 02 FEBC CME15 BSR **4
FEB8 BD 00 FEBC BSR **2
FEB8 39 RTS

*
FEBD C6 B0 A CME16 LDB ##B0
FEBF 86 0A A CME20 LDA ##0A NOT READY ?
FEC1 C5 B0 A BITB ##B0
FEC3 26 0E FED3 BNE CME30 YES
FEC5 86 0C A LDA ##0C SEEK ERROR
FEC7 C5 10 A BITB ##10
FEC9 26 08 FED3 BNE CME30 YES
FECB 86 0D A LDA ##0D CRC ERROR ?
FECD C5 08 A BITB ##08
FECF 26 02 FED3 BNE CME30 YES
FED1 86 0F A CME25 LDA ##0F TIME OVER ERROR
FED3 1A 01 CME30 SEC
FED5 39 RTS

*
FED6 D6 18 A FREADY EQU * DSTAT
FED8 54 A LDB LSRB
FED9 24 04 FEDF BCC YESRDY READY ? FDC READY CHECK.
FEDB C6 D0 A LDB ##D0 FORCE INTERRUPT FDC NOT READY.
FEDD D7 18 A STB DCMD FORCE INTERRUPT COMMAND.

* FDC READY
FEDF 96 18 A YESRDY LDA DDATA
FEE1 43 COMA
FEE2 97 18 A STA DDATA
FEE4 BD D2 FEB8 BSR CME15
FEE6 91 18 A CMPA DDATA
FEE8 26 CD FEB7 BNE CME11
FEEA CE 5000 A LDU ##5000
FEEB 33 5F A RDYCHK LEAU -1,U TIME OUT ?
FEEF 1183 0000 A CMPU #0
FEF3 27 06 FEBF BEQ TIMOUT YES TIMEOUT Z=1 ZERO.
FEF5 D6 18 A LDB DSTAT
FEF7 2B F4 FEED BMI RDYCHK Z=0 NON ZERO SET.
FEF9 C6 01 A LDB #1
FEFB 39 TIMOUT RTS

***** DRIVE SELECT
*
FEFC A DRSEL EQU *
*FDB MSKF0 FIR0,IR0 MASK. (B2/01/20).
FEFC D6 1D A LDB DDRIE READ CURRENT DRIVE.
FEFE F7 FFE5 A STB DMOTOR
FF01 C4 03 A ANDB ##03 MSB=MOTOR FLAG 1=ON 0=OFF.
FF03 F7 FFE0 A STB CRDRIV CURRENT DRIVE NO SET.
FF06 CE FFE1 A LDU #CURTRK CURRENT HEAD POSITION SAVE AREA.
FF09 A6 07 A LDA RCBUNT,X
FF0B 84 03 A ANDA ##03 DESTINATION UNIT SELECT.
FF0D B7 FFE8 A STA DSTDRV
FF10 7D FFE5 A TST DMOTOR WAS ON ?
FF13 2B 17 FF2C BMI WASON TES, IT WAS MOTOR ON.
* NO MOTOR MUST BE ON AND WAIT 1 SEC AND CLEAR THE TRACK POSITION SAVE
* AREA, FORCE RESTORE ALL DRIVES.
*
FF15 B6 B0 A DVTCLR LDA ##B0
FF17 34 42 A RESTLP PSHS A,U DRIVE NO SAVE.
FF19 BD FE60 A JSR RRESEN CALL RESTORE ROUTINE.
FF1C 35 42 A PULS A,U DRIVE NO PULL DOWN.
FF1E 4C INCA
FF1F B1 84 A CMPA ##B4 ALL END ?
FF21 26 F4 FF17 BNE RESTLP NO, CONTINUE.

*
FF23 B6 FFE8 A LDA DSTDRV
FF26 BA B0 A ORA ##B0
FF28 97 1D A STA DDRIE DESTINATION DRIVE SET.
FF2A 20 09 FF35 A BRA CHKRDY

*
FF2C B1 FFE0 A WASON CMPA CRDRIV DESTINATION=CURRENT ?
FF2F 27 04 FF35 BEQ CHKRDY YES
FF31 BA B0 A ORA ##B0
FF33 97 1D A STA DDRIE IT IS NOT SAME, DRIVE REG SET.

*
FF35 BD 9F FED6 BSR FREADY READY CHECK.
FF37 27 03 FF3C BEQ NOTRDY NOT READY.
FF39 1C FE CLC NORMAL
FF3B 39 RTS

*
FF3C 1A 01 NOTRDY SEC NOT READY.

```

## リスト3 バブル・モード(BANK 1)アセンブル・リスト

```

FF3E 39          RTS
          *
FF3F CE FF3F A TM1S EQU * 1 SECOND TIMER=985
FF42 8D 03D9 A LDU #03D9
FF44 33 09 FF4D TM1S5 BSR TM1M
FF46 11B3 5F A LEAU -1,U
FF4A 26 0000 A CMFU #0
FF4C 39 FF42 A BNE TM1S5
          RTS

FF4D 10BE 0099 A TM1M LDY #99 1 MS TIMER.
FF51 31 3F A TM1M5 LEAY -1,Y DEC Y
FF53 26 FC FF51 A BNE TM1M5
FF55 39          RTS
          *
FF56 74 04 A TM20M PSHS B
FF58 C6 14 A LDB #20
FF5A 8D F1 FF4D TM20LP BSR TM1M
FF5C 5A A DECB
FF5D 26 FB FF5A A BNE TM20LP
FF5F 35 94 A PULS B,PC

*****
          *
FF61          A RDWRIT EQU * SECTOR WRITE.
FF61          A RDREAD EQU *
FF61 8D FE7F A JSR SEEK DRIVE SELECT ,SEEK TRACK
FF64 26 5A FF60 A BNE DREAD?
          *
FF66 1A50 A FDB MSKFI0 IR0,FIR0 MASK.(B2/01/20).
          *
FF68 A6 06 A LDA RCBSID,X
FF6A 84 01 A ANDA #1 GET UNIT.
FF6C 97 1C A STA DSIDE
FF6E A6 05 A LDA RCBSCT,X
FF70 97 1A A STA DSECTR SECTOR SET
FF72 A6 84 A LDA RREQ,X GET REQUEST NO.
FF74 81 0A A CMPA #FDREAD DREAD ?
FF76 27 1B FF90 BSW DREADX YES ,BRANCH DREAD.
FF78 B6 A0 A LDA #5A0 WRITE COMMAND SET
FF7A 97 18 A STA DCMD
FF7C 10AE 02 A LDY RCBDDB,X
FF7F A6 A0 A DWRT1 LDA 0,Y+ DATA GET
FF81 D6 1F A DWRT2 LDB DDR0 DR0 ?
FF83 2A 04 FF89 BPL DWRT5 NO
FF85 97 1B A STA DDATA DATA STORE
FF87 20 F6 FF7F A BRA DWRT1

FF89 58          *
FF8A 2A F5 FF81 A DWRT5 ASLB EXTRACT COMMAND END TO MSB.
FF8C 8D BF FF4D BPL DWRT2
FF8E 20 14 FFA4 BSR TM1M
          RWEND
          *
          *
FF90          A DREADX EQU * SECTOR READ
FF90 86 80 A LDA #80 READ COMMAND SET
FF92 97 18 A STA DCMD
FF94 10AE 02 A LDY RCBDDB,X
FF97 D6 1F A DREAD1 LDB DDR0 DR0 ?
FF99 2A 06 FFA1 BPL DREAD5 NO
FF9B 96 1B A LDA DDATA LOAD DATA
FF9D A7 A0 A STA 0,Y+
FF9F 20 F6 FF97 A BRA DREAD1
          *
FFA1 58          *
FFA2 2A F3 FF97 A DREAD5 ASLB EXTRACT COMMAND END TO MSB.
FFA4 D6 18 A RWEND BPL DREAD1
FFA6 26 02 FFAA LDB DSTAT ERROR ?
FFA8 4F A BNE DREAD5
FFA9 39          RTS
          *
FFAA 86 0A A DREAD6 LDA #0A
FFAC C5 80 A BITB #80 NOT READY ?
FFAE 26 10 FFC0 BNE DREAD?
FFB0 4C A INCA
FFB1 C5 40 A BITB #40
FFB3 26 0B FFC0 BNE DREAD?
FFB5 4C A INCA
FFB6 C5 14 A BITB #14
FFB8 26 06 FFC0 BNE DREAD?
FFBA 4C A INCA
FFBB C5 0B A BITB #0B
FFBD 26 01 FFC0 BNE DREAD?
FFBF 4C A INCA
FFC0 1A 01 DREAD? SEC
FFC2 39          RTS
          *
FFC3 3B          * NMIRT1 RTI DO NOTHING.
          * BOOT ROM END ADDRESS=$FFDF
          *
FFFE          ORG $FFFE
FFFE FE00 A FDB $0000 BOOT ENTRY ADDRESS.
          *

```

## リスト 4 DOSモード(BANK 2)アセンブル・リスト

```

FE00                                ORG    $FE00
                                     *
                                     TTL    5 INCH OS MODE(NO.2) BOOTSTRAP LOADER (B2 03 01).
                                     *
FE00 20  FE00 09  FE0B  *  BOTENT EQU  *  5 INCH OS MODE.
                                     *  STARTS
                                     *
FE02 7E  FE0F  A  BOTST* EQU  $FC0F
FE05 7E  0300  A  IPLADR EQU  $0300
FE08 7E  FE0B  A  RSTOR  JMP  RRESTR
FE0B 7E  FF71  A  DWRITE JMP  RDWRIT
FE0B 7E  FF71  A  DREAD  JMP  RDREAD
                                     *
FE0B CC  FE0B  A  STARTS EQU  *
FE0E FD  FF03  A  LDD     LDD     *NMIRTI NMI VECTOR SET.
FE11 B6  FD    A  STD     PNMI
FE13 1F  BB    A  LDA     LDA     *$FD
                                     *  DIRECT PAGE=FD.
FE15 10CE FE15  A  DIS*V EQU  *  TRY TO READ IPL CODE.
FE19 96  05    A  LDS     LDS     *BOTST*
FE1B 44  *      A  LDA     LDA     SBOOUT
FE1C 25  2F  FE4D BDC     BDC     FDERR
FE1E BE  FE53  A  LDX     LDX     *BOTRCB RCB ADDRESS FOR BOOT.
FE21 BD  10  FE33 BSR     BSR     DREADZ FLOPPY READ ROUTINE.
FE23 4D  *      A  TSTA    TSTA    ANY ERROR ?
FE24 26  27  FE4D BNE     BNE     FDERR
FE26 BE  FE5B  A  LDX     LDX     *BOTRCB
FE29 BD  0B  FE33 BSR     BSR     DREADZ
FE2B 4D  *      A  TSTA    TSTA    ANY ERROR ?
FE2C 26  1F  FE4D BNE     BNE     FDERR
FE2E 1F  BB    A  TFR     A+DP DP=0
FE30 7E  0300  A  JMP     JMP     IPLADR
                                     *
FE33 C6  05    A  DREADZ EQU  *  5 TIMES RETRY.
FE35 34  04    A  XXREAD PSHS  #5
FE37 BD  CF  FE0B BSR     BSR     DREAD
FE39 35  04    A  PULS    B
FE3B 4D  *      A  TSTA    TSTA
FE3C 27  03  FE41 BE9     BE9     XXRTS
FE3E 5A  *      A  DECB    DECB
FE3F 26  01  FE42 BNE     BNE     ZXRTY
FE41 39  *      A  XXRTS  RTS
FE42 34  14    A  ZXRTY  EQU  *
FE44 BE  FE63  A  PSHS    PSHS  X,B
FE47 BD  B9  FE02 BSR     BSR     RETRCB
FE49 35  14    A  PULS    PULS  X,B
FE4B 20  E8  FE35 BRA     BRA     XXREAD
                                     *
FE4D B6  B1    A  FDERR  LDA  #B1
FE4F 97  03    A  STA    STA    BELL
FE51 20  FE  FE51 BRA     *
                                     *
FE53 0A  FE53  A  BOTRCB EQU  *
FE54 00  0A    A  FCB     FCB     FDBREAD R0N0
FE55 0300 0A    A  FCB     FCB     0 STATUS
FE57 00  0A    A  FCB     FCB     0 IPLADR DATA STORE ADDRESS.
FE5B 01  0A    A  FCB     FCB     0 TRACK=0
FE59 00  0A    A  FCB     FCB     1 SECTOR=1
FE5A 00  0A    A  FCB     FCB     0 SIDE=0
FE5A 00  0A    A  FCB     FCB     0 UNIT=0
                                     *
FE5B 0A  FE5B  A  BOTRCB EQU  *
FE5C 00  0A    A  FCB     FCB     FDBREAD R0N0 SET.
FE5D 0400 0A    A  FCB     FCB     0 STATUS.
FE5F 00  0A    A  FCB     FCB     0 IPLADR+$100 DATA STORE ADDRESS.
FE60 02  0A    A  FCB     FCB     0 TRACK=0
FE61 00  0A    A  FCB     FCB     2 SECTOR=2
FE62 00  0A    A  FCB     FCB     0 SIDE=0
FE62 00  0A    A  FCB     FCB     0 UNIT=0
                                     *
FE63 0B  0B    A  RETRCB FCB  FDBREAD
FE64 0000 0A    A  FDB     FDB     0,0,0
FE66 0000 0A    A
FE68 0000 0A    A
FE6A 00  0A    A  FCB     FCB     0 UNIT=0
                                     *
                                     TTL    5 INCH FD DRIVER MODULE(F-BASIC : ROM-BASED MODE.)
                                     *
                                     ***** TYPE 1 COMMAND
                                     *
FE6B CE  FE6B  A  RRESTR EQU  *  SEEK TRACK 0.
FE6E A6  FF01  A  LDU     LDU     *BCURTRK
FE6E A6  07    A  LDA     LDA     #RCURNT.X
                                     *  BUG CORRECTION 2 --> 2S VERSION B2/01/11 SOURCE UPDATE.
FE70 B4  03    A  RRESEN EQU  *  RRESEN LABEL POSITION WAS CORRECTED. CLR A+U -> ANDA #003.
FE72 6F  06    A  ANDA    ANDA    #003 UNIT SELECT.
FE74 D6  1D    A  CLR     CLR     A+U TRACK ADDRESS 0 SET.
FE76 BA  1D    A  LDB     LDB     *DDRIVE
FE78 97  1D    A  ORA     ORA     #00
FE7A BD  4C  FE0B BSR     BSR     CME15
FE7C 96  1D    A  STA     STA     *DDRIVE
FE7E 2A  4D  FE0D BPL     BPL     CME16
FE80 5D  *      A  TSTB    TSTB
FE81 2B  0B  FE8B BRL     BRL     Z1000
FE83 96  02    A  LDA     LDA     #2
FE85 BD  FF4F  A  JSR     JSR     TW15
FE88 4A  *      A  DECA    DECA
FE89 26  FA  FE85 BNE     BNE     Z500
FE8B B6  0A  FE85 LDA     LDA     #00A RESTORE COMMAND (U=0)
FE8D 20  1A  FE89 BRA     BRA     CMEXEC BRANCH COMMAND EXECUTION

```

316

## リスト 4 DOSモード(BANK 2)アセンブル・リスト

```

FF45 0D 9F FEE6 CHKRDY BSR FREADY READY CHECK.
FF47 27 03 FF4C BSR NOTRDY NOT READY.
FF49 1C FE CLC NORMAL
FF4B 39 RTS

FF4C 1A 01 NOTRDY SEC NOT READY.
FF4E 39 RTS

FF4F CE FF4F A TM1S EQU * 1 SECONUD TIMER=985
FF52 0D 03D9 A LDU ##03D9
FF54 33 09 FF5D TM1S5 BSR TM1M
FF56 1183 5F A LEAU -1,U
FF5A 26 0000 A CMPLU #0
FF5C 39 F6 FF52 BNE TM1S5
RTS

FF5D 10BE 0099 A TM1M LDY ##99 1 MS TIMER.
FF61 31 3F A TM1M5 LEAY -1,Y DEC Y
FF63 26 FC FF61 BNE TM1M5
FF65 39 RTS

FF66 34 04 A TM20M PSHS B
FF68 C6 14 A LDB #20
FF6A 0D F1 FF5D TM20LP BSR TM1M
FF6C 5A DEC B
FF6D 26 FB FF6A BNE TM20LP
FF6F 35 B4 A PULS B,PC

*****
FF71 BD FF71 A RDWRIT EQU * SECTOR WRITE.
FF74 26 5A FF71 A RDREAD EQU *
FF74 26 5A FF71 A JSR SEEK DRIVE SELECT ,SEEK TRACK
FF74 26 5A FF71 A BNE DREAD?

FF76 1A50 A FDB MSKF19 IR0,FIR0 MASK. (B2/01/20).

FF7B A6 06 A LDA RCBSID,X
FF7A B4 01 A ANDA #1 GET UNIT.
FF7C 97 1C A STA DSIDE
FF7E A6 05 A LDA RCBSCT,X
FF80 97 1A A STA DSECTR SECTOR SET
FF82 A6 B4 A LDA RQNG,X GET REQUEST NO.
FF84 B1 0A A CMPLA #FDREAD DREAD ?
FF86 27 18 FFA0 BSR DREADX YES ,BRANCH DREAD.
FF88 B6 A0 A LDA ##A0 WRITE COMMAND SET
FF8A 97 18 A STA DCMD
FF8C 10AE 02 A LDY RCBDDBA,X
FF8F A6 A0 A DWRT1 LDA O,Y+ DATA GET
FF91 D6 1F A DWRT2 LDB DDR0 DR0 ?
FF93 2A 04 FF99 BPL DWRT5 NO
FF95 97 1B A STA DDATA DATA STORE
FF97 20 F6 FFBF BRA DWRT1

FF99 5B DWRT5 ASLB EXTRACT COMMAND END TO MSB.
FF9A 2A F5 FF91 BPL DWRT2
FF9C 0D BF FF5D BSR TM1M
FF9E 20 14 FFB4 BRA RWEND

*****
FFA0 B6 FFA0 A DREADX EQU * SECTOR READ
FFA2 97 B0 A LDA ##B0 READ COMMAND SET
FFA4 10AE 02 A STA DCMD
FFA7 D6 1F A DREAD1 LDB DDR0 DR0 ?
FFA9 2A 06 FFB1 BPL DREAD5 NO
FFAB 96 1B A LDA DDATA LOAD DATA
FFAD A7 A0 A STA O,Y+
FFAF 20 F6 FFA7 BRA DREAD1

FFB1 5B DREAD5 ASLB EXTRACT COMMAND END TO MSB.
FFB2 2A F3 FFA7 BPL DREAD1
FFB4 D6 1B A RWEND LDB DSTAT ERROR ?
FFB6 26 02 FFBA BNE DREAD6
FFB8 4F CLRA
FFB9 39 RTS

FFBA B6 0A A DREAD6 LDA ##0A
FFBC C5 B0 A BITB ##B0 NOT READY ?
FFBE 26 10 FFD0 BNE DREAD7
FFC0 4C 40 A INCA #0B
FFC1 C5 0B FFD0 BITB ##40 WRITE PROTECTED ?
FFC3 26 0B FFD0 BNE DREAD7
FFC5 4C 14 A INCA #0C
FFC6 C5 06 FFD0 BITB ##14 RECORD NOT FOUND,LOST DATA ?
FFC8 26 06 FFD0 BNE DREAD7
FFCA 4C 0B A INCA #0D CRC ERROR ?
FFCB C5 01 FFD0 BITB ##0B
FFCD 26 01 FFD0 BNE DREAD7
FFCF 4C 01 FFD0 INCA #0E DELETED DATA MARK DETECTED !
FFD0 1A 01 DREAD7 SEC SET CARRY FOR DRSEL ERROR INTERFACE.
FFD2 39 RTS

FFD3 3B NMIRTI RTI DO NOTHING.
* BOOT ROM END ADDRESS=$FFDF
*
FFFE FE00 A ORG $FFFE
FFFE FDB BOTENT BOOT ENTRY ADDRESS.

```



## リスト 5 8" DOSモード(BANK 3)アセンブル・リスト

```

FE00          ORG    $FE00
              *
              *      TTL      B INCH      DOS MODE(NO 3) BOOT STRAP LOADER (B2 03 20).
              *
FE00 20      FE00  A BOTENT EQU    *      FM-B BOOT PROGRAM OS MODE (NO 3)
              15  FE17  BRA      *      STARTS
              *
              02EF  A BOTSTK EQU    $02EF  STACK POINTER SET.
              0300  A IPLADR EQU    $0300
              02F0  A BOTRCB EQU    $02F0  RCB ADDRESS.
              *
FE02 7E      FE02  A DRLOCK JMP    RDRLOC  DOOR LOCK (8 INCH)
FE05 7E      FE05  A SETMOD JMP    RSETMO  MODE SET (8 INCH)
FE08 7E      FE08  A DISK2  JMP    RDISK2  DISK2 SENSE (8 INCH)
FE0B 7E      FE0B  A RESTRB JMP    RRESTX  RESTORE (8 INCH)
FE0E 7E      FE0E  A WTSDEN JMP    RWTSD  SINGLE DENSITY SINGLE SECTOR WRITE (8 INCH)
FE11 7E      FE11  A WTDEN  JMP    RWTDD  DOUBLE DENSITY SINGLE SECTOR WRITE (8 INCH)
              *
FE14 7E      FE14  A DMAERR JMP    XDMAER  DMA DEADLOCK ERROR.
              *
              FE0E  A RDSDEN EQU    WTSDEN  SINGLE DENSITY SINGLE SECTOR READ (8 INCH)
              FE11  A RDDDEN EQU    WTDEN  DOUBLE DENSITY SINGLE SECTOR READ (8 INCH)
              *
              FE17  A STARTS EQU    *
FE17 CC      FFD7  A      LDD    #NMIRTI NMI VECTOR SET.
FE1A FD      FFFC  A      STD    PNMI
FE1D 86      FD    A      LDA    #$FD
FE1F 1F      8B    A      TFR    A,DP      DIRECT PAGE=FD.
              *
              FE21  A DISKV  EQU    *      TRY TO READ IPL CODE.
FE21 10CE    02EF  A      LDS    #BOTSTK
FE25 96      05    A      LDA    SBSOUT
FE27 44      *
FE28 25      4A    FE74  LSRA
              FE2A  A FDBOOT EQU    *
              *
              *      B INCH FLOPPY INITIALIZE
              *
              *      IF THE POWER ON RESET OR RESET SIGNAL CAUSE FD INITIAL SET ,THE FOLLOWING
              *
              *      STEPS ARE NOT NECESSARY FOR THIS PROGRAM.
              *
              *      LDD    #0      INITIALIZE COMMAND SET.
              *      LBSR   REGSET  SELECT COMMAND REG.
              *
              *
FE2A 7F      FFE9  A      CLR    LSIZE  SECTOR LENGTH=128 BYTE.
              *      CALL SETMOD
              *
              *      LDX      #SMDRCB
              *      BSR      RSETMO  CALL SETMOD ROUTINE
              *      TSTA     ERROR CHECK
              *      BNE      FDERRX  ERROR !
              *
              *      DMA REGISTER INITIAL SET.
              *
              *      RCR,ICR,DCR SET
              *
              *      LDA      #X00001000 PCR DATA PATARN : CH3 TXEN3=1
              *      STA      DMAPCR  PCRSET
              *
              *      LDA      #X00001000 DCR DATA PATAR : 4 CHANNEL MODE
              *      STA      DMADCR
              *      CLR      DMAICR  #X00000000 ALL CHANNEL IRQ MASK. ICR SET.
              *
              *
              *      TRACK 0,SECTOR 1 - 4,UNIT 0,SIDE 0 GET.
              *
FE2D BE      02F0  A      LDX    #BOTRCB RCB ADDRESS FOR BOOT.
FE30 86      22    A      LDA    #34    8 INCH FLOPPY SINGLE DENSITY SECTOR READ.
FE32 A7      84    A      STA    RQNG,X  REQUEST
FE34 CC      0300  A      LDD    #IPLADR DMA BUFFER ADDRESS SET.
FE37 ED      02    A      STD    RCBDBA,X
FE39 CD      0001  A      LDD    #00001 TRACK=0 SECTOR=1
FE3C ED      04    A      STD    RCBTRK,X TRACK,SECTOR SET.
              *      LDD    #0      SIDE=0 UNIT=0
FE3E 5F      *      CLR    ASSUME ACCA=0,CLEAR ACCB, ACC D = $0000.
FE3F ED      06    A      STD    RCBSD,X SIDE,UNIT SET
              *
              *      FLOPPY DISK READ ( 5 TIMES RETRY )
              *
              *
FE41 C6      FE41  A CNTRED EQU    *
FE43 34      05    A      LDB    #5      RETRY COUNT=5.
FE45 8D      14    A      A XXREAD PSHS  B,X
FE47 35      C7    FE0E  BSR    RDSDEN  CALL SINGLE DENSITY SECTOR READ.
FE49 4D      14    A      PULS  B,X
FE4A 27      13    FE5F  TSTA     NORMAL ?
FE4C 81      0C    FE74  BEQ     YES NORMAL.
FE4E 26      24    FE74  CMPA    #12    FLOPPY HARD ERROR ?
              BNE     FDERRX  NON RECOVERBLE ERROR.
              *
              *      ERROR OCCURED
              *
FE50 5A      *      DECB

```

## リスト 5 8" DOSモード (BANK 3) アセンブル・リスト

```

FE51 27 21 FE74 BE9 FDERRX ERROR RETURN. RETRY COUNT OVER.
*
* EXECUTE RETRY
*
FE53 34 14 A PSHS B,X
FE55 BE FE7A A LDX #RETRCB
FE58 17 00B1 FEDC LBSR RRESTX CALL RESTORE 8 INCH.
FE5B 35 14 A PULS B,X
FE5D 20 E4 FE43 BRA XXREAD

FE5F EC FE5F A READOK EQU *
FE61 C3 02 A LDD RCDBA,X
FE64 ED 02 A ADDD #12H ADD SECTOR LENGTH.
FE66 6C 05 A STD RCDBA,X
FE68 E6 05 A INC RCBSCT,X SECTOR=SECTOR+1
FE6A C1 05 A LDB RCBSCT,X
FE6C 25 D3 FE41 CMPB #5 SECTOR 1,2,3,4 READ ?
FE6E 4F BLD CNTRED CONTINUE
FE6F 1F BB A CLR A DP=0
FE71 7E 0300 A JMP IPLADR

FE74 86 FE74 A FDERRX EQU *
FE76 97 B1 A LDA #B1 FD ERROR.
FE7B 20 FE FE7B STA BELL LET RING AN ALARM.
* SOFT HALT.
*
* MDRCB SPC 5
* FCB 29 RQND=29 : SETMOD.
* FCB 0 STATUS
* FCB 1 PRE ONE TRACK READ MODE.
* FCB 0 128 / SECTOR

FE7A 1F A RETRCB FCB 31 8 INCH RESTORE REQUEST NO.
FE7B 0000 A FDB 0,0,0
FE7D 0000 A
FE7F 0000 A
FEB1 00 A FCB 0 UNIT 0
*
* TTL 8 INCH FD DRIVER MODULE. (A PART OF BIOS) (B2 03 20)
*
*** 8 INCH FLOPPY DISK DRIVER MODULE.
*
* REGISTER DESCRIPTION
*
* (WRITE)
* RRR EQU $FD30 REGISTER RESERVE REGISTER.
* RDR EQU RRR+1 REGISTER DATA REGISTER.
* WDR EQU RRR+2 DATA REGISTER.
* ... EQU RRR+3 RESERVED.
*
* (READ)
* SSR EQU RRR+0 IFDC SYSTEM STATUS REGISTER.
* FSR EQU RRR+1 FDC(MB8B66 OR MB8B76) STATUS REGISTER.
* RER EQU RRR+2 DATA REGISTER.
* ... EQU RRR+3 RESERVED.
*
* IFDC SYSTEM STATUS REGISTER.
* B7: IFDC READY
* B6-B5: RESERVED
* B4: TRANSFER ACKNOWLEDGE
* B3: DMA ERROR
* B2: DATA ERROR
* B1: COMMAND ERROR
* B0: FDD ERROR
*
* (IFDC INTERNAL REGISTER)
*
0000 A CMDREG EQU 0 COMMAND REGISTER
0001 A TRKREG EQU 1 TRACK REGISTER
0002 A SSREG EQU 2 SECTOR START REGISTER
0003 A SEREG EQU 3 SECTOR END REGISTER
0004 A SIDREG EQU 4 SIDE REGISTER
0005 A DNSREG EQU 5 DENSITY REGISTER
0006 A DRVREG EQU 6 DRIVE REGISTER
0007 A DLCREG EQU 7 DOOR LOCK REGISTER
0008 A ERESET EQU 8 ERROR RESET
0009 A SFORMAT EQU 9 SECTOR FORMAT DATA
000A A SECDAT EQU 10 SECTOR LENGTH DATA
000B A MODREG EQU 11 MODE SET REGISTER
*
* (SECTOR LENGTH)
* SINGLE DOUBLE
* 0 : 128/26
* 1 : 256/15 256/26
* 2 : 512/8 512/15
* 3 : ----- 1024/8
*
*****
* (IFDC COMMAND DESCRIPTION)

```

## リスト 5 8" DOSモード(BANK 3)アセンブル・リスト

```

0000      *
0000      A ZINIT EQU 0 IFDC INITIALIZE
0001      A ZFORMAT EQU 1 MAKE SOFT SECTOR FORMAT
0002      A ZSEEK EQU 2 SEEK TRACK
0003      A ZLOAD EQU 3 READ SECTOR (FLOPPY --) IFDC BUFFER)
0004      A ZSAVE EQU 4 WRITE SECTOR(FLOPPY <-- IFDC BUFFER)
0005      A ZREADP EQU 5 DATA GET (PROGRAM TRANSFER) FMB <-- IFDC
0006      A ZWRITP EQU 6 DATA PUT (PROGRAM TRANSFER) FMB <-- IFDC
0007      A ZREADD EQU 7 DATA GET (DMA TRANSFER) FMB <-- IFDC
0008      A ZWRITD EQU 8 DATA PUT (DMA TRANSFER) FMB <-- IFDC
0009      A ZLDRDP EQU 9 DIRECT READ SECTOR (PROGRAM) FMB <-- FLOPPY
000A      A ZWTSVP EQU 10 DIRECT WRITE SECTOR (PROGRAM) FMB --> FLOPPY
000B      A ZLDRDD EQU 11 DIRECT READ SECTOR (DMA) FMB <-- FLOPPY
000C      A ZWTSVD EQU 12 DIRECT WRITE SECTOR (DMA) FMB --> FLOPPY
000D      A FDTEST EQU 13 FLOPPY DISK SEEK TRACK OPERATION.
      *
0000      A SINGLE EQU 0 SINGLE DENSITY
0001      A DOUBLE EQU 1 DOUBLE DENSITY
      *
0003      A DMAWTP EQU %00000011 HALT BASTE MODE, ADDRESS=ADDRESS+1, MEMORY TO I/O.
0002      A DMARDP EQU %00000010 * ,I/O TO MEMORY.
      *
      * DOOR LOCK ROUTINE(8 INCH FLOPPY)
      *
FE82      A RDRLOC EQU *
      * LBSR SPACT STACK POINTER SAVE & IFDC ACTIVE CHECK.
FE82 10FF FFEA A STS SPSAVE STACK POINTER SAVE.
FE86 A6 07 A LDA RCBUNT,X GET UNIT NO.
FE88 B4 03 A ANDA #$03 B7-B2 CLEAR.
      *
FE8A 6D 02 A TST RCBLOC,X DOOR LOCK ?
FE8C 27 02 FE90 BEQ DROPEN NO,DOOR UNLOCK.
      *
FE8E BA 80 A ORA #$80 YES,DOOR LOCK.
FE90 C6 07 A LDB #DLCREG SELECT DOOR LOCK REGISTER.
FE92 17 00F4 FF89 LBSR REGSET
FE95 4F A CLRA NORMAL RETURN.
FE96 39 A XXRTS RTS
      *
      * READ MODE SET ROUTINE(8 INCH FLOPPY)
      *
FE97      A RSETMO EQU *
      * LBSR SPACT STACK POINTER SAVE & IFDC ACTIVE CHECK.
FE97 10FF FFEA A STS SPSAVE STACK POINTER SAVE.
FE9B A6 02 A LDA RCBMOD,X
FE9D B1 FF A CMPA #$FF MODE SET ?
FE9F 27 05 FE96 BEQ SETLN NO,DON'T SET THE MODE.
      *
      * FLOPPY DISK READ MODE SET
      *
FEA1 C6 08 A LDB #MODREG SELECT MODE SET REGISTER.
FEA3 17 00E3 FF89 LBSR REGSET
      *
FEA6 A6 03 A SETLN EQU * SECTOR LENGTH SET.
FEA8 B1 FF A LDA RCBSLN,X
FEAA 27 08 FEB4 CMPA #$FF SECTOR LENGTH SET ?
      *
      * FLOPPY DISK SECTOR LENGTH SET
      *
FEAC C6 0A A LDB #SECDAT SELECT SECTOR REGISTER.
FEAE 17 00DB FF89 LBSR REGSET
FEB1 B7 FFE9 A STA LSIZE SECTOR LENGTH MEMO AREA,USE AT READ/WRITE.
FEB4 4F A SETRTS CLRA NORMAL RETURN
FEB5 39 A RTS
      *
      * DISK2 SENSE ROUTINE(8 INCH FLOPPY)
      *
FEB6      A RDISK2 EQU *
      * LBSR SPACT STACK POINTER SAVE & IFDC ACTIVE CHECK.
FEB6 10FF FFEA A STS SPSAVE STACK POINTER SAVE.
FEBA A6 07 A LDA RCBUNT,X GET UNIT NO.
FEB8 BA 03 A ANDA #$03 B7-B2 CLEAR.
FEBE BA 40 A ORA #$40 DOOR OPEN/CLOSE BIT INHIBIT.
FEC0 C6 07 A LDB #DLCREG SELECT DOOR LOCK REGISTER.
FEC2 17 00C4 FF89 LBSR REGSET
      *
FEC5 96 31 A LDA FSR
      *
FEC7 E6 07 A LDB RCBUNT,X
FEC9 C4 03 A ANDB #$03 B7-B2 CLEAR.
FECB C1 03 A RDLOOP CMPB #3
FECD 27 04 FED3 BEQ DK2BIT
FECF 48 A LSLA
FED0 5C A INCB
FED1 20 FB FEB3 BRA RDLOOP
      *
FED3 5F A DK2BIT CLRB ASSUME DISK 2
FED4 4D A TSTA DISK 2 ?
FED5 2A 01 FEDB BPL FLGSET YES DISK 2
      *
FED7 5C A INCB NO,NOT DISK 2
FED8 E7 02 A FLGSET STB RCBDK2,X MEDIA FLAG SET
FEDA 4F A CLRA
FEDB 39 A RTS
      *
      * RESTORE (SEEK TRACK 0) ROUTINE (8 INCH FLOPPY)
      *
FEDC      A RRESTX EQU *
      * LBSR SPACT STACK POINTER SAVE & IFDC ACTIVE CHECK.
FEDC 10FF FFEA A STS SPSAVE STACK POINTER SAVE.
FEE0 A6 07 A LDA RCBUNT,X UNIT NO SET.
FEE2 C6 06 A LDB #DRVREG

```

## リスト 5 8" DOSモード (BANK 3) アセンブル・リスト

```

FEE4 17 00A2 FF89 * WHEN SEEK TRACK IS EXECUTED WITH VERIFY, THIS CODING IS ESSENTIAL.
* CLRA DENSITY=0
* LDR #DNSREG
* LBSR RESET

FEE7 0C 0001 A LDD #0+256+TR*REG TRACK NO=0, TRACK REGISTER SELECT.
FEFA 17 009C FF89 LBSR RESET SEEK TRACK 0
FEED 86 02 A LDA #ZSEEK
FEFF 20 2D FF1E BRA ENTRST

* SINGLE DENSITY SECTOR WRITE (8 INCH FLOPPY)

FEF1 A RWTSD EQU *
* BSR SPACT STACK POINTER SAVE & IFDC ACTIVE CHECK.
* STS SPSAVE STACK POINTER SAVE.
FEF1 10FF FFEA A STS SPSAVE STACK POINTER SAVE.
FEF5 17 00B4 FF7C LBSR STSDEN SET SINGLE DENSITY..
FEFB 20 06 FF00 BRA WTCOMN BRANCH WRITE SECTOR COMMON PROGRAM.

* DOUBLE DENSITY SECTOR WRITE (8 INCH FLOPPY)

FEFA A RWTDE EQU *
* BSR SPACT STACK POINTER SAVE & IFDC ACTIVE CHECK.
* STS SPSAVE STACK POINTER SAVE.
FEFA 10FF FFEA A STS SPSAVE STACK POINTER SAVE.
FEFE 8D 7F FF7F BSR STDDEN SET DOUBLE DENSITY.
FF00 8D 61 FF63 WTCOMN BSR PARSET SET DISK PARAMETER.
FF02 8D 2E FF32 BSR SETDMA DMA ADDRESS & COUNTS SET & IRQ, FIRQ MASK.

FF04 A6 84 A LDA RQNO,X GET REQUEST NO.
FF06 81 0A A CMPA #10 5 INCH FLOPPY SECTOR READ ?
FF08 27 0E FF18 BEQ REDCMD YES
FF0A 81 2E A CMPA #34 8 INCH FLOPPY SINGLE DENSITY SECTOR READ ?
FF0C 27 0A FF18 BEQ REDCMD YES
FF0E 81 23 A CMPA #35 8 INCH FLOPPY DOUBLE DENSITY SECTOR READ ?
FF10 27 06 FF18 BEQ REDCMD YES

* WRITE SECTOR ( RQNO=9,32 OR 33 )

FF12 C6 03 A LDB #DMAWTP DMA CONTROL REGISTER SET
* STA C3CHCR MEMORY TO I/O

FF14 86 0C A LDA #ZWTSVD DIRECT WRITE SECTOR COMMAND (DMA)
FF16 20 04 FF1C BRA DMAENB BRANCH DMA ENABLE.

* READ SECTOR ( RQNO=10,34 OR 35 )

FF18 FF18 A REDCMD EQU *
LDB #DMARDP DMA CONTROL REGISTER SET
* STA C3CHCR I/O TO MEMORY

FF1A 86 0B A LDA #ZLDRDD DIRECT READ SECTOR COMMAND (DMA)
FF1C D7 53 A DMAENB STB C3CHCR
FF1E 5F ENTRST CLRB SELECT COMMAND REGISTER.
FF1F 97 31 A STA RDR COMMAND SET.
FF21 D7 30 A STB RRR DMA START.

* DMA START NEED 10 MICRO SEC.

FF23 96 30 A DMSACK LDA SSR GET IFDC SYSTEM STATUS REGISTER.
FF25 2B FC FF23 BMI DMSACK WAIT FOR IFDC BUSY.

FF27 96 30 A WTDLOP LDA SSR GET IFDC SYSTEM STATUS REGISTER.
FF29 2A FC FF27 BPL WTDLOP IF BUSY, BRANCH WTDLOP.

FF2B 84 0F A LDA SSR GET STATUS.
FF2D 1026 00B7 FFBB ANDA #Z0000111: ANY ERROR ?
LBNE ERRORP YES, ERROR OCCURED.
* CLRA FOR DMA END NORMAL RETURN.
FF31 39 RTS RETURN

*
* *****
* DMA PARAMETER SET
*
* 1. DMA ADDRESS SET
*
* 2. DMA BYTE COUNTS SET
*
* 3. IRQ & FIRQ MASK
*
* *****

FF32 EC FF32 A SETDMA EQU *
FF34 DD 02 A LDD RCBDDBA,X GET SECTOR BUFFER ADDRESS
4C A STD CH3ADR DMA ADDRESS SET.

FF36 F6 FFE9 A LDB LSIZE SECTOR DATA GET. 0=128,1=256,2=512,3=1024
FF39 4F 01 A CLRA ACC A=0 SECTOR LENGTH=128*2^N
FF3A 1A BEKI SEC C=1
FF3C 49 ROLA LNGGET MAKE 2^N (N=ACC B)
FF3D 5D TSTB
FF3E 27 03 FF43 BEQ XMULTI
FF40 5A DECB
FF41 20 F9 FF3C BRA BEKI

*
* ACC A=2^N
* ACC B=128
*
FF43 C6 FF43 A XMULTI EQU *
80 A LDB #128 GET 128 IN ACC B
FF45 3D A MUL SECTOR LENGTH=ACCA*ACCB (2^N*128)
FF46 DD 4E A STD CH3BCR SET DMA BYTE COUNTS.

```

## リスト 5 8" DOSモード (BANK 3) アセンブル・リスト

```

FF4B D6 54 A LDB DMAPCR
FF4A C4 77 A ANDB #$77
FF4C CA 0B A ORB #$0B
FF4E D7 54 A STB DMAPCR

FF50 D6 55 A *
FF52 C4 07 A LDB DMAICR
FF54 CA 00 A ANDB #$07
FF56 D7 55 A ORB #$00
STB DMAICR

FF5B D6 56 A *
FF5A C4 77 A LDB DMADCR
FF5C CA 0B A ANDB #$77
FF5E D7 56 A ORB #$0B
STB DMADCR

FF60 1A50 A *
FDB MSKF10 IR0,FIR0 MASK.

FF62 39 *
RTS RETURN
*
*
* IFDC PARAMETER SET ROUTINE
*
* INPUT -- X=RCB ADDRESS
*
*
* OUTPUT -- X=RESERVED
*
*
*
FF63 A PARSET EQU *
04 A LDA RCBTRK,X TRACK NO GET.
FF65 C6 01 A LDB #TRKREG SELECT TRACK REGISTER.
FF67 BD 20 FF89 BSR REGSET
FF69 A6 05 A LDA RCBST,X SECTOR NO SET.
LDB #SSREG SECTOR START REG SELECT.

FF6B 5C *
FF6C BD 1B FF89 INCB REGSET
BSR REGSET SECTOR END REG SELECT.
FF6E 5C INCB
FF6F BD 1B FF89 BSR REGSET
FF71 A6 06 A LDA RCBSD,X GET SIDE DATA.
LDB #SIDREG SELECT SIDE REG.

FF73 5C *
FF74 BD 13 FF89 INCB
BSR REGSET SIDE SET.
FF76 A6 07 A LDA RCBUNT,X GET DRIVE NO.
FF7B C6 06 A LDB #DRVREG SELECT DRIVE REG.
FF7A 20 0D FF89 BRA REGSET UNIT SET.
*
*
*
FF7C FF7C A STSDEN EQU *
00 A LDA #SINGLE SINGLE DENSITY SET.
FF7E BC A FCB SKIP2 CMPX # : SKIP 2 BYTE.
*
FF7F FF7F A STDDEN EQU *
01 A LDA #DOUBLE DOUBLE DENSITY SET.
FF81 C6 05 A LDB #DNSREG SELECT DENSITY REGISTER.
FF83 20 04 FF89 BRA REGSET DENSITY SET.
*
*
*
*
* STACK POINTER SAVE & IFDC ACTIVE CHECK.
*
FF85 A SPACT EQU *
* LEAU 2,S STACK POINTER SAVE.
* INC SPSAVE SPSAVE=SPSAVE+2
* INC SPSAVE 2-BYTE INCREASE FOR ADJUSTMENT OF SPACT ITSELF
SUBROUTINE CALL.
* EXECUTION GO TO FOLLOWING STEP.
*
* IFDC ACTIVE CHECK.
*
* COMMON ROUTINE 1
*
* REGISTER ACTIVE CHECK ROUTINE.
*
FF85 A ACTIVE EQU *
* LDA #$55
* STA RDR SS PATARN ACTIVE CHECK.
* CMPA RDR
* BNE NONACT
* COMA
* STA RDR AA PATARN ACTIVE CHECK.
* CMPA RDR
* BNE NONACT
* CLRA
* RTS NORMAL RETURN. IFDC ACTIVE.

FF85 0A A NONACT EQU *
0A A LDA #10 DISK NON AVAILABLE ERROR.
FF87 20 34 FF8D BRA ERRRTS RETURN TO BIOS MAIN PROGRAM.
*
* COMMON ROUTINE 2
*
* REGISTER DATA SET ROUTINE
*
* ACC A -- REGISTER DATA
* ACC B -- REGISTER NO
*
FF89 A REGSET EQU *

```

```

FFB9 97 31 A STA RDR REGISTER DATA SET.
FFBB D7 30 A STB RRR REGISTER SELECT.
*
*
* IFDC READY CHECK.
*
* IF NORMAL -- RETURN BY RTS.
* IF ABNORMAL -- RETURN TO BIOS MAIN.
* ACC A,B,X=PRESEVED
*
FFBD 34 FFBD A RDYCK0 EQU *
16 A PSHS A,B,X
FFBF 4F CLR A FD BUSY (READY=0) CHECK.
FF90 8D 0C FF9E BSR CKLOOP
* FCB SKIP2 CMPX #: SKIP 2-BYTES
*
*RDYCK1 EQU *
* PSHS A,B,X
FF92 86 80 A LDA #$B0 FD READY (READY=1) CHECK.
FF94 8D 0B FF9E BSR CKLOOP SURE READY
FF96 96 30 A LDA SSR GET STATUS.
FF98 84 0F A ANDA #00001111 ANY ERROR ?
FF9A 26 1C FFBB BNE ERRORP YES,ERROR OCCURED.
FF9C 35 96 A PULS A,B,X,PC
*
* IFDC READY-BUSY,ONLINE CHECK ROUTINE.
*
FF9E 8E FF9E A CKLOOP EQU *
09 A LDX #4 ABOUT 1.23S * 4 TIMES
FFA1 10BE FFFF A CKLOP0 LDY $FFFF
FFA5 31 3F A CKLOP1 LEAY -1,Y TIME OVER ?
FFA7 26 06 FFAF BNE CKLOP2
FFA9 30 1F A LEAX -1,X
FFAB 27 0B FFBB BSR NONACT YES TIME OVER. SET DISK NOT READY ERROR NO.
FFAD 20 F2 FFA1 BRA CKLOP0
*
FFAF 1F 89 A CKLOP2 TFR A,B ACCA => ACCB
FFB1 0B 30 A EOR B SSR
FFB3 04 80 A ANDB #$B0 BIT7:EQUAL ?
FFB5 26 EE FFA5 BNE CKLOP1 NO,CONTINUE
FFB7 39 RTS
*
* IFDC ERROR HANDLING.
* INPUT ---- BIOS MAIN RTS ADDRESS IS SAVED IN SPSAVE.
*
FFBB 84 FFBB A ERRORP EQU * ERROR DETECTED.
09 A ANDA #00001001 DMA OR FDD ERROR ?
FFBA 26 0A FFC6 BNE FDERR YES FDD OR DMA
*
* LDA #1 YES RCB PARAMETER ERROR.
* INCA ACC A +=1.
FFBC 4C 0B A ERRRTS LDB #ERRSET ERROR FLAG RESET.
FFBD 06 30 A STB RRR
FFBF 07 30 A RETMAN LDS SPSAVE GET DIRECT RTS ADDRESS TO BIOS MAIN.
FFC1 10FE FFEA RTS
FFC5 39
*
FFC6 06 FFC6 A FDERR EQU * 11 INCH FLOPPY DISK HARD ERROR
31 A LDR FSR GET FDC STATUS REGISTER.
FFC8 86 0A A LDA #10 ASSUME DRIVE NOT READY.
FFCA 05 80 A BITB #$B0 DRIVE NOT READY ?
FFCC 26 EF FFB3 BNE ERRRTS YES,NOT READY.
FFCE 4C 0B A INCA ACCA+=1 ASSUME WRITE PROTECT.
FFCF 05 40 A BITB #$40 WRITE PROTECT ?
FFD1 26 EA FFB3 BNE ERRRTS YES,WRITE PROTECT.
FFD3 86 0C A XDMAER LDA #12 ACCA+=12 FLOPPY DISK HARD ERROR.
*
* PCR,ICR,DCR SET.
*
*LDR #00001000
*STB DMAPCR PCR SET.
*STB DMADCR DCR SET.
*CLR DMAIDR ICR SET.
*
* OR DMA DEADLOCK ERROR.
FFD5 20 E6 FFB3 BRA ERRRTS DDM,RECORD NOT FOUND,SEEK ERROR,CRG ERROR LOST DATA.
*
* NMI VECTOR SET
*
FFD7 0D FFD7 A NMIRTI EQU * PROCESS ROUTINE.B2/01/18 ADDITION.
56 A TST DEADLK DMA DEADLK ?
FFD9 2B F8 FFD3 BMI XDMAER YES,DMA DEADLOCK.
FFDB 3B RTI NO,MEMORY PARITY ERROR.
*
* BOOT ROM MEMORY STAMP.
*
FFDD 82 A ORG $FFDD
FFDE 03 A FCB $B2 YEAR
FFDF 20 A FCB $03 MONTH
* FCB $20 DAY
*
* RESTART VECTOR SET.
*
FFFE FFE0 A ORG $FFFE
FDB BOTENT

```

## F M-8活用研究正誤表

## ☆ 6809の特徴

p.6 インデックス・スタック・ポインタ操作命令

$$CP \rightarrow CMP$$

## ☆拡張版Kコンパイラ

p.18 高速ソート・プログラム

```

540 *MID; (*GET MIDDLE VALUE *)
550 * MIDWRK1:=1;MIDWRK2:=2;MIDWRK3:=3
560 * IF MIDWRK2>MIDWRK3 THEN EXCHANGE(ADR(MIDWR
K2),ADR(MIDWRK3));FI
570 * IF MIDWRK1>MIDWRK2 THEN EXCHANGE(ADR(MIDWR
K1),ADR(MIDWRK2));FI
580 * IF MIDWRK2>MIDWRK3 THEN EXCHANGE(ADR(MIDWR
K2),ADR(MIDWRK3));FI
590 * MID=MIDWRK2
600 *RETURN

```

## ☆p. 31 スクリーン・アセンブラ

```

1070 TAS=" ":FOR I=0 TO 15:D=PEEK(PM+I):T=T+" "+
FNS(D):AY(J)=AY(J)+D:AX(I)=AX(I)+D:IF (D<32)+(127
D)+(247D) THEN TS=TS+"*":NEXT LE TS=TS+CHR$(
D):NEXT
1100 IF (J MOD 4=0) THEN PRINT
1910 LOCATE 61+HX#2,3+HY+HY#4:IF (D<32)+(127=D)+(
247D) THEN S="*": ELSE S=CHR$(D)
1990 FOR I=0 TO 15:AX(I)=0:AY(I)=0:NEXT:ZS=0:RET
URN

```

## ★2PASS DISASSEMBLER

p.43 \$6780～\$74FFまでは、アセンブラ・ダンプ・リスト(p.37～p.38)の\$4780～\$54FFと同じです。

## ☆6809デバツガ

p.45コマンド表 Sセーブ Sファイル名, 先頭アドレス,  
最終アドレス, 実行開始アドレス **RETURN**

p.47 DAD4, 74E4番地09を06に訂正.

## ☆マシン語データ↔配列変換ルーチン

p. 104 360.....LOCATE60,12.....→LOCATE60, 20

テレビ黒板

p.111 マシン語ダンフ・リスト

7D84 FC  $\rightarrow$  7C

## ☆フローチャート・ジェネレータ

p.124 360.....LEFT\$ (L\$, 1) = "12EM" .....

$$\rightarrow \text{LEFT\$}(\text{L\$}, 3) = \text{"REM"} \dots\dots$$

☆TEXT CONVERTER

p. 139 改良点

78E0 EE EG→EF

AUTO-2001

p.147 CMT-1 プログラム・リスト

```
2615 IF A$="7" THEN GOSUB 2810:HARDC:GOSUB 11000
:FOR J=0 TO 4:PRINT#2:NEXT J:GOTO 99
2624 IF A$="7" THEN HARDC:GOTO 99
```

## ☆日本語ワード・プロセッサの制作

p. 160 1370  $\dot{\text{H}}$  & H3020  $\rightarrow$  & H30211410  $\dot{\nu}$  & H4720  $\rightarrow$  & H4728

1460 11 &amp; H2576 → &amp; H2577

## ●グラフィック花札こいこい（初版）

p. 210 ダンプ・リスト

563C~563E番地は7B, 63, C6

## ☆汎用I/Oポートの製作（初版）

p. 218の右側下から4行めは「穴あきベーク」です。

### ☆PSGカードの製作

p. 226 PSGカード回路図 (ト図)

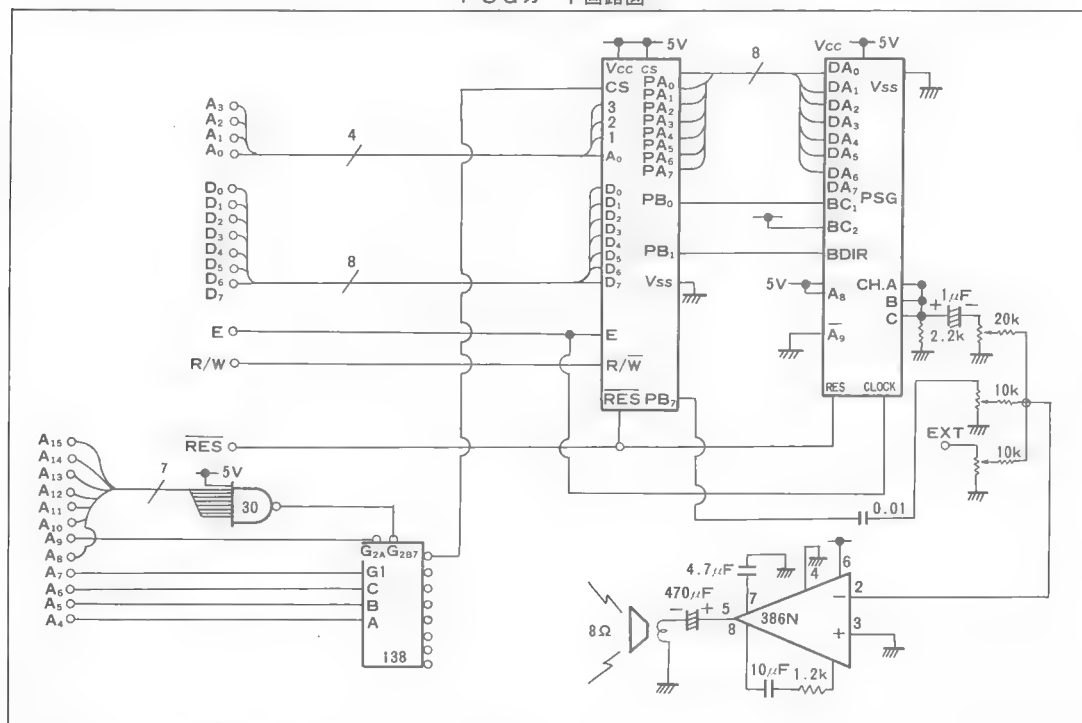
## ☆ SUPER TRACER (初版, 第2版)

p. 48 “主な特色+ただの特色”のところで上から1行め

図 1 EA<sup>注1</sup>、(EA)両ストックのTOP 1WORDは

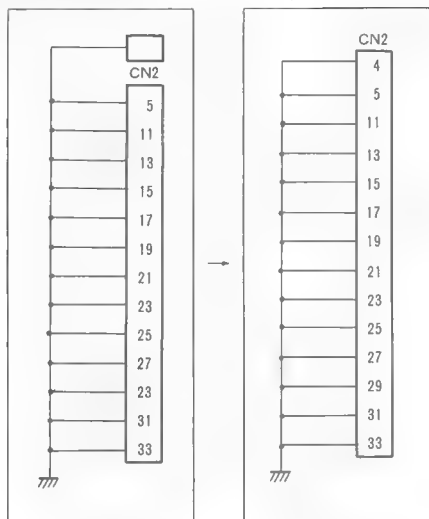
正 EA, EA<sup>注</sup>, S および U 両スタックのトップ 2 バイトは,

PSGカード回路図

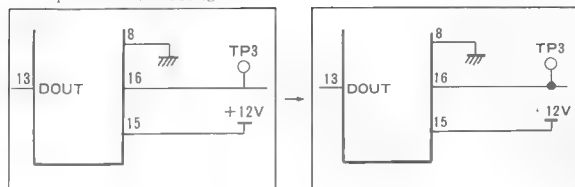


## ★全回路図集〔初版〕

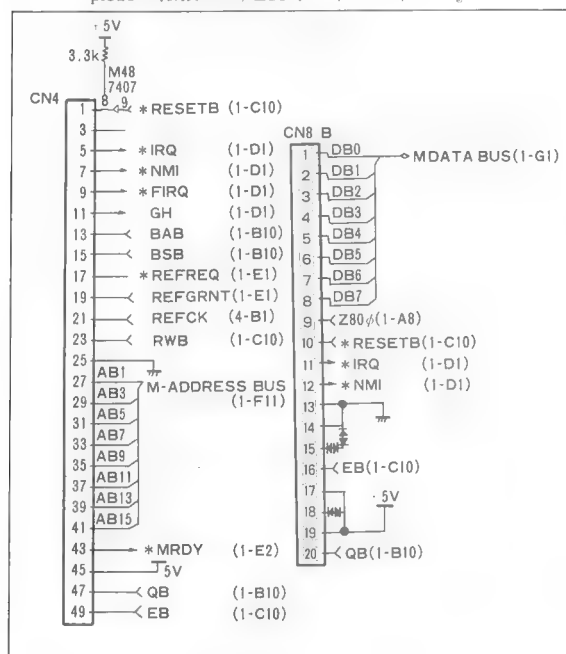
・ p.325 『ライン・プリンタ/リニア』



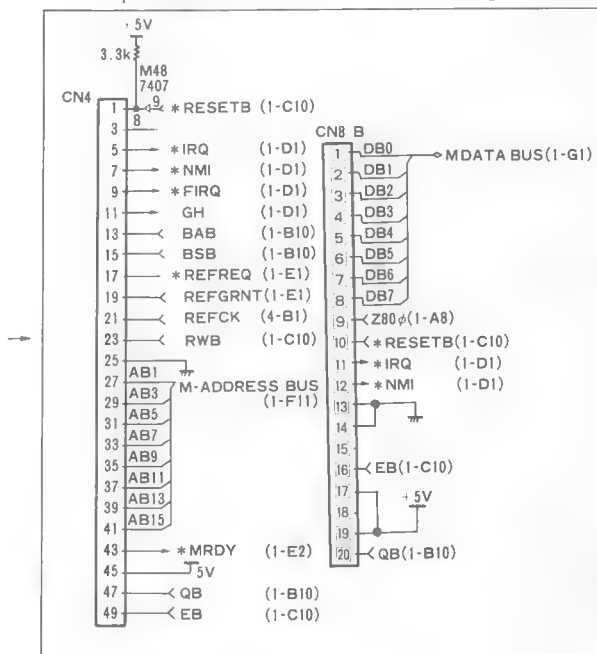
・ p.326 『A/D変換』



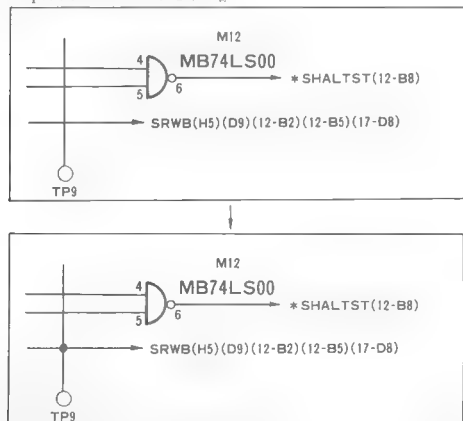
・ p.328 『拡張バス/Z80カード・コネクタ』



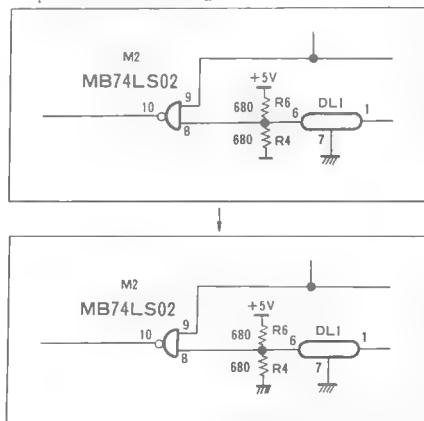
・ p.328 『拡張バス/Z80カード・コネクタ』



・ p.331 『サブCPU(1)』

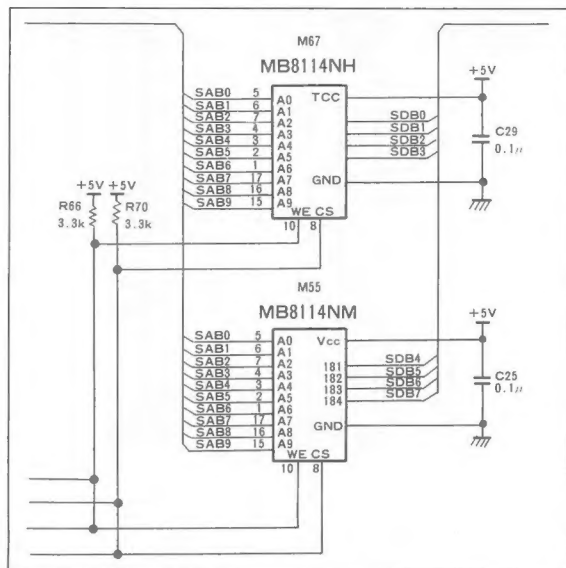
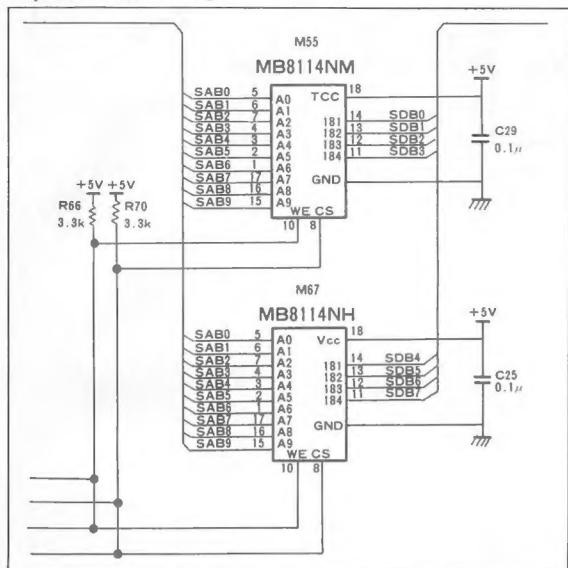


・ p.347 『Z80カード』

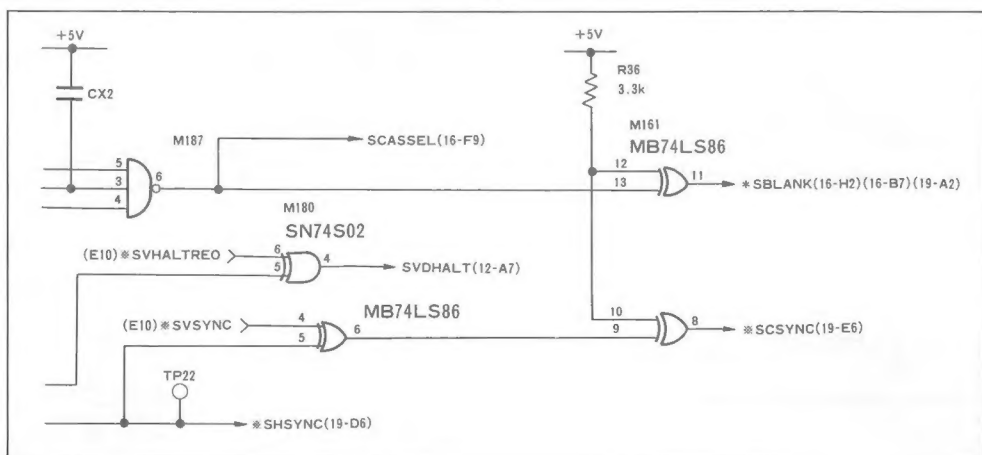
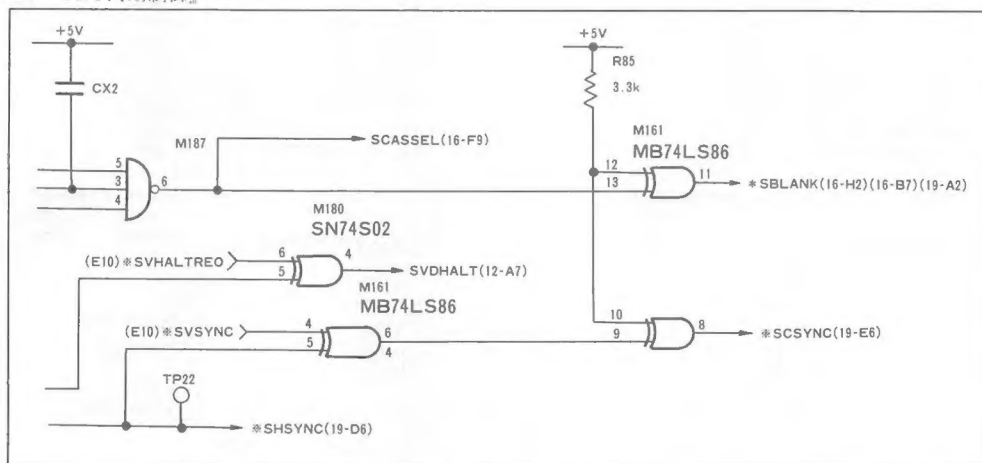




・p.335 『共有メモリ』



・p.337 『CRT同期制御』



# FM-SOFT

## INFORMATION

### カセット・サービス

本誌に掲載されているプログラムは、コンパクト  
よりカセット・テープで販売いたします。

### B 5判ブック型パッケージ

商品番号	商品名	機種名	内 容	定 価 (送料別)
608	文庫秀麗Ver.2.0	FM-8	本格的日本語ワード・プロセッサ	¥9,800
813	拡張Kコンパイラ	FM-7/8	6809の特徴を生かした整数型コンパイラ	¥4,500
829	エディタ・アセンブラV.2.0	FM-7/8	フルRAMで動くエディタが強化されたアセンブラ	¥4,500
840	F-BASIC EXPANDER	FM-7	F-M-8用をF-M-7に移植	近日発売
CD-1019	サウンド・エディタVer.2.0	FM-7/8	PSG用楽符入力音楽演奏プログラム	近日発売

### P ケースパッケージ

商品番号	商品名	内 容	I/O掲載 (年月号)	定 価 (送料別)
------	-----	-----	----------------	--------------

### FM-8

244	地底最大の作戦	地底基地に攻め込むヒーローをやつつけろ	—	¥3,000
246	ブラック・ホール	3-Dグラフィック・プログラム	—	¥3,000
254	グラフィック麻雀	バイの色までソックリ、麻雀ゲームの決定版	'81.12	¥3,000
255	簡易モニタ	F-M-8に使い易いモニタ	'81.12	¥3,000
253	逆アセンブラ	L 3 版を移植	—	¥3,000
289	平安京エイリアン	移植版	—	¥3,000
310	Z80カード逆アセンブラ	Z80の逆アセンブラ・プログラム	'82.2	¥3,000
317	F-M-8有象無象	いろいろなソフトが入ってます	'82.3	¥3,000
309	音声入出力プログラム	F-M-8が言葉をしゃべる	'82.2	¥3,000
339	N-F-BASICコンバータ	P-Cの豊富なソフトをF-Mに!	'82.4	¥3,000
427	サウンド・コンパイラ	効果音、音楽用データの編集に!	'82.6	¥3,000
426	漢字SYMBOL&&BASIC良話	漢字をSYMBOL文で拡大します	'82.6	¥3,000
591	2バス逆アセンブラ	ラベル付で318のソースをジェネレート	F M活研	¥3,000
592	6809デバッグ	MZのS.R.D.を目標に作ったデバッグ	F M活研	¥3,000
593	6809トレース	プログラム開発の強力なツール	F M活研	¥3,000
594	Z80コミュニケーション	6809とZ80の汎用入出力ルーチン	F M活研	¥3,000
595	Z80モニタ	Z80カードを120%活かすツール	F M活研	¥6,500
596	テレビ黒板	パターン作成用ツール	F M活研	¥3,000
609	F-BASIC EXPANDER	F-BASICとモニタのコマンドを拡張	'82.9	¥3,000
622	スクリーン・アセンブラ	逆アセンブラ+アセンブラ	F M活研	¥3,000
625	ウォーク・ゲーム	転ばないように、闇の中を歩く	—	¥3,000
641	独立モニタ	フルRAMで動くモニタ	'82.11	¥3,000
642	3次元迷路	3次元迷路は2次元迷路に飽きた人用	'82.11	¥3,000
693	イエローサブマリン	PCGカードがあれば音も出ます	T 14	¥3,000
695	CRASH	オール・マシン語のハイレゾゲーム	T 14	¥3,000
744	MEMORY CHANGER	フルRAMで動くチェックサムエディタ	'82.12	¥3,000
760	ザ・キューティ	Mr. L/Oのキューティ狩り	'83.1	¥3,000
762	変数リスト	デバッグ、改造に便利な変数のリストを!	'83.1	¥3,000
775	ALL RAM DISK BASIC	DOSモードでDISK BASICを!	'83.2	¥3,000
777	THE 女学生 追草編	落ちてくるリンプを受ける	'83.2	¥3,000
796	F-Mラリー	サブCPUを使う高速ゲーム	'83.3	¥3,000
825	U-MONITOR	モニタを拡張してブレーク・ポイントも	'83.6	¥3,000

商品番号	商品名	内 容	I/O掲載 (年月号)	定 価 (送料別)
------	-----	-----	----------------	--------------

### FM-7/8

776	実戦 株式チャート	選定、年足を表示し、株の動きを予測する	'83.2	¥3,000
243	1人ホーカゲーム	「神格が因大なる本」を参考に作られました	—	¥3,000
256	ミニ&ディスタアセンブラ	L 3 の移植版	'81.12	¥3,000
262	グラフィック麻雀(漢字ROM使用)	254の漢字ROM版	'81.12	¥3,000
286	グラフィック花札こいこい	ドット単位の色指定を使い、画面は感動的	—	¥3,000
287	グラフ作成プログラム	技術者、学生に	'82.1	¥3,000
288	日本語ワード・プロセッサNo1	漢字ROMとK-505を使用	'81.12	¥3,000
290	スノーキ	移植版	—	¥3,000
307	ALL CAST STAR TREK	本物そっくりのキャラクター!	'82.2	¥3,000
308	マシン語→DATA文作成プログラム	マシン語がBASICのDATA文に!	'82.2	¥3,000
316	カラーパターン・エディタ	キャラクターを自在に作って...	'82.3	¥3,000
319	SPACE FIGHTER	気分は戦闘機乗り	'82.3	¥3,000
320	ザ・麻雀	好評のザ・麻雀をF-Mに移植!	T 12	¥3,000
341	日本語ワード・プロセッサNo2	漢字ROM付のシステムで文書作成	'82.4	¥3,000
358	囲碁研修プログラム	PC-8001の移植版	—	¥3,000
359	易占プログラム	PC-8001の移植版	—	¥3,000
414	フローチャート・ジェネレーター	人のアルゴリズムも簡単に理解できる	'82.5	¥3,000
384	SPACE TRASEPORTER	相手をおびきよせて...	T 12	¥3,000
385	Submarine WARS	絵を見るような美しさ!	T 12	¥3,000
457	BLACK JACK	ドット・グラフィックを使った本格派	—	¥3,000
245	激平五日	白と黒のチップが裏と表で攻防戦	—	¥3,000
340	SPACE SHOOTER	侵略者を撃ち落とせ!	'82.4	¥3,000
598	Auto Plot	グラフ作成のツール	F M活研	¥3,000
598	AUTO-2001 FM-8	好評のAUTO-2001のFM版です	F M活研	¥4,500
579	UFO WARS	現われては消えるUFOを撃て	T 13	¥3,000
580	アニメ付アナログ時計	朝がつかないあなたに	T 13	¥3,000
623	シーソーゲーム	シーソーを使って黒船制り	'82.10	¥3,000
635	金星基地SOS	金星基地を守るのは君の暗算力?	T 14	¥3,000
636	THE 女学生 下校編	オオカミに捕まらないように	T 14	¥3,000
694	棋盤プログラム	棋譜の登録、再生ができる	T 14	¥3,000
828	キンチョマン(キー版)	宇宙人撃退には昆虫系キンチョン君を	'83.5	¥3,000
CD-1020	ALIEN FALL	衝撃波を避けエイリアンを撃つ	7/8活研	¥3,000
CD-1021	SUPER MONITOR	モニタアセンブラ+逆アセンブラ+トレース	7/8活研	近日発売
CD-1022	スーパーパターン・エディタ	上下左右のパターンが一度に見られます	7/8活研	近日発売
CD-1023	BASIC ANALYZER	カラーリスト、変数リストなどデバッグ用	7/8活研	近日発売
CD-1024	日本語ワード・プロセッサ	BASIC+マシン語で横60文字まで使えます	7/8活研	近日発売
CD-1025	戦車最前戦	戦車隊にかこまれた、道を突破できるか!	—	¥3,000

★お求めは全国の有名書店、マイコン・ショップへ、または直接、コンパクトまでお申し込みください。(郵送手数料1本500円、2本以上1本増すごとに100円増)

発売元 **COMPAC INC.** (株)コンパクト / 〒151 東京都渋谷区代々木1-37-1 ぜんらくビル5F ☎(03)375-3401

## 編集後記

■FM活用研究第2弾『FM-7/8活用研究』、やっと出たとみなさんお思いでしょうね。私もやっと出せたと思っています。なにせ、記事を選ぶのに時間を取られ、雑用に時間を割かれてもう大変な騒ぎでした。

たぶん、読者のみなさんはサブCPUの使い方がわからず、その利用法の詳しい説明を欲しがっていると思います。そこで今回はその方面の記事を多く載せましたので参考にしてください。

本誌の進行を無視して、エンジン・ルームではサブCPUの使い方を着々と“もの”にしています。本誌には間に合いませんでしたが、とてつもないものができています。これはKコンパイラ+αで達成しています。

KコンパイラはFMの強力な味方です。Kコンパイラの文法がわからないと思う方、FMの文法書と見比べてください。あまり変わらないでしょう。

ディスプレイ・サブシステムの使い方がわからない方、本誌のプログラムを解析してください。必ずディスプレイ・サブシステムの使い方が理解できます。初めは“猿まね”でも良いじゃないですか。それでわかれば、後は焼いて食おうが煮て食おうが好き勝手にできるのです。

小さなプログラム、“RANDOM BOX”も良く見てください（本当はこども、マシン語はアセンブル・リストにしたかった）。ここにも思わぬ発見があるはずですよ。

本誌はKコンパイラを重視していますが、これはKコンパイラのRUNTIMEルーチンが小さいためです。800バイト足らずのRUNTIMEルーチンなんて、そうざらにありません。基本サブルーチンしかサポートしていないか

らだ、と言ってしまうばそれまでですが、でも、Kコンパイラ・グラフィック・サブルーチンを使えば、グラフィックも簡単に扱えます（Kコンパイラでここまで扱えるようにしたユーザーのみなさんの力量には感嘆いたします）。どの行にサブルーチンを使ってもかまわず、ラベル名だけとなるためですね。

くれぐれも注意しましょう。Kコンパイラはラベル名もしくは関数名で使い、行番号では動きません！また、変数名のチェックもお忘れなく。

本誌ではあるメーカーのプリンタだけになってしまいましたが、他のメーカーのものでもコントロール・コードを換えることで動作するはずですよ。同じメーカーのものでもコントロール・コードの違いで動作しないのですから。

(K<sub>1</sub>)

■何をするにも、情熱や能力を持ち続けることは、事を成功させる大きな要因だと思います。それだけでなく向上させなくては、とおっしゃる読者諸氏も多いでしょうが、何十年単位で考えると、なかなか難しいことと思いませんか。

1冊の本を作るにしても、とりかかってからで上がるまでは、さまざまなトラブルは起こるし、イライラが続いたり、体調を崩したりで、エネルギーを常に充填させていることはちょっと不可能。それが長丁場ともなればなおのことです。

さて、本書の出来はどうでしょうか。本の出来具合の評価は、読者と編集者との間でどのようなひらきがあるのかと考えてみました。たとえば1つの間違いを、前者はなんだこんな間違いををしてと思い、後者はとてもここまで目が届かなかったと思うのではないのでしょうか（いいわけをしています）。

(Y<sub>2</sub>)

## I/O別冊 FM-7/8活用研究

昭和58年7月25日 初版発行

©1983

編集 I/O編集部

発行人 星 正明

発行所 株式会社工学社

〒151 東京都渋谷区代々木1-37-1 ぜんらくビル5F

☎(03)375-5784代〔営業〕

☎(03)320-1218代〔編集〕

振替 東京5-22510

定価2,500円

印刷：(株)耕文社



息も  
つか  
せぬ  
盛り  
あ  
が  
り。



富士通の興奮パソコン<FM-7>。発売以来、もうクライマックスの連続です。豊富な機能のハードに興奮、ますます充実するソフトに感激。ハードがよいから、ソフトもいい。価格が手頃、と話題集中。全国の青少年を、ビジネスマンを大いに盛りあがらせています。

#### 感激のソフトが、ますます充実

<FM-7>で使えるソフトが、どんどんふえています。いろいろなゲームや、ホビー、ビジネス、教育、話題集中のワープロなど、市販のソフトはもとより、<FM-8>用の流通ソフトもその多くが、そのまま使えます。しかも、本体内に実装され、盛り沢山の機能をサポートしているF-BASICにより、プログラミングも実に容易です。

#### 簡易言語がついてく

うれしいことに<FM-7>には簡易言語が標準装備されています。家庭では住所録や家計簿、またオフィスでは各種資料の作成など実に幅広く利用できます。作表や計算、検索や分類、ファイリングなどが自由自

在。データを大きい順から並べかえたり(ソート機能)、必要な条件に合ったデータだけ拾いだしたり(サーチ機能)して、表形式のファイルにまとめることができます。

●お手持ちのカラーTVに接続可能です。(オプションのTVアダプタを使用)。

●サウンド機能で、ゲーム効果音や8オクターブ、三重和音までの音楽演奏もOK。

●カラーグラフィック画面は640×200ドットの高分解能。ドット毎に8色まで色指定でき、パレット機能で色交換も簡単です。

●漢字ROMカード(オプション)を装着すれば、日本語ワープロとしても使えます。

先端技術が夢中にさせる興奮パソコン。

# FM-7

¥126,000

(本体価格  
簡易言語ソフト付)

セブン

ビジネス用途を大きく広げる高級パソコン。

## FM-11

イレブン

EX ¥398,000

(本体価格・簡易言語ソフト付)

AD ¥338,000

(本体価格・簡易言語ソフト付)

ST ¥268,000

(本体価格)

高級ホビーからビジネスまでの多オパソコン。

## FM-8

エイト

¥218,000

(本体価格)

富士通株式会社 半導体統轄営業部 〒105 東京都港区虎ノ門2-3-13 ☎(03)502-0161

●札幌営業所(011)271-4311 ●東北営業所(0222)64-2131 ●長野営業所(0262)26-8222 ●静岡営業所(0542)54-9131

●名古屋営業所(052)201-8611 ●大阪営業所(06)344-1101 ●広島営業所(082)221-2288 ●九州営業所(092)411-6311

マイコンスカイラブ

FMシリーズのハードからソフトまで一挙に展示実演。あなたのパソコンのコンサルタントとしてご利用ください。

●東京・虎ノ門(03)591-1091/591-2561 ●東京・秋葉原(03)251-1448/251-1449 ●大阪(06)344-7628/341-0486 ●名古屋(052)221-6016 ●札幌(011)241-4185 ●広島(082)247-3949 ●仙台(0222)66-8711

富士通

定価2500円